



**Universidade do Minho**

DEPARTAMENTO DE ENGENHARIA INFORMÁTICA

UNIVERSIDADE DO MINHO

MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA

## LABORATÓRIOS DE INFORMÁTICA 4



*Autores:*

Paulo LIMA

Pedro MACHADO

João AZEVEDO

Paulo ARAÚJO

Hugo CUNHA

ANO LETIVO 2019-2020

---

## RESUMO

Este relatório foi elaborado no âmbito do desenvolvimento de um assistente de *Fitness*, uma aplicação multi plataforma para um ginásio que ajuda e guia os utilizadores na sua rotina diária de exercícios e o seu plano alimentar, estabelecidos pelo seu instrutor.

A primeira etapa compreendeu com a apresentação do domínio da aplicação, bem como a apresentação do sistema que nos propusemos a desenvolver. De seguida, foi desenvolvido o Modelo de Domínio do sistema, os seus Use Cases e, por último, um plano de futuro desenvolvimento nas suas próximas fases.

Os objectivos que se seguiram marcaram o início do processo de materialização das ideias e expectativas apresentadas. Este início foi marcado com o desenvolvimento do modelo Conceptual e Lógico da Base de Dados, especificação de Use Cases, criação de *Views* para a ferramenta a utilizar e o Diagrama de Classes do sistema.

Para além destas tarefas indicadas, o grupo explorou também várias tecnologias para a implementação da aplicação.

**Área de Aplicação:** Serviços de assistência de *Fitness*.

**Palavras-Chave:** Engenharia de Software, Ginásio, Microsoft, Base de Dados.

---

# CONTEÚDO

<b>1</b>	<b>Resumo</b>	<b>2</b>
<b>2</b>	<b>Introdução</b>	<b>6</b>
2.1	Contextualização . . . . .	6
2.2	Apresentação do Caso de Estudo . . . . .	6
2.3	Motivação e Objetivos . . . . .	7
2.4	Estrutura do Relatório . . . . .	7
<b>3</b>	<b>Levantamento de Requisitos</b>	<b>8</b>
3.1	Requisitos Funcionais . . . . .	8
3.1.1	Autenticação de utilizadores . . . . .	8
3.1.2	Criação planos de treino e alimentares . . . . .	9
3.1.3	Consultar planos de treino e alimentares . . . . .	9
3.1.4	Criação de metas . . . . .	9
3.1.5	Marcar presença num espaço[EXTRA] . . . . .	9
3.1.6	Apresentação de lotação dos diferentes espaços do ginásio[EXTRA]	10

3.1.7	Consultar estatísticas de lotação por horário[EXTRA] . . . . .	10
3.2	Requisitos Não Funcionais . . . . .	11
<b>4</b>	<b>Modelo de Domínio</b>	<b>12</b>
<b>5</b>	<b>Diagrama de Use Cases</b>	<b>14</b>
<b>6</b>	<b>Especificação de Use Cases</b>	<b>15</b>
6.1	Consultar Aulas de Grupo . . . . .	15
6.2	Consultar Evolução . . . . .	15
6.3	Consultar Plano Alimentar . . . . .	16
6.4	Consultar Plano de Treino . . . . .	16
6.5	Consultar resultado da última avaliação de um dado utilizador . . . . .	17
6.6	Consultar Resultado da última avaliação . . . . .	18
6.7	Consultar Última Avaliação . . . . .	18
6.8	Criar Plano alimentar para Cliente . . . . .	19
6.9	Criar Plano de treino para Cliente . . . . .	20
6.10	Criar resultado de avaliação para Cliente . . . . .	20
6.11	Criar utilizador . . . . .	21
6.12	Editar planos de aulas semanal . . . . .	22

6.13	Log In . . . . .	23
6.14	Log Out . . . . .	24
6.15	Marcar presença na aula . . . . .	24
6.16	Remover Utilizador . . . . .	25
<b>7</b>	<b>Arquitetura da Camada de Negócios</b>	<b>26</b>
7.1	Camada de Negócios . . . . .	26
7.2	Dicionário das principais Classes . . . . .	26
7.3	Descrição da arquitetura . . . . .	27
7.4	Diagrama de Classes . . . . .	28
<b>8</b>	<b>Camada de Dados</b>	<b>29</b>
8.1	Modelo Lógico . . . . .	29
8.2	Diagrama do Modelo Lógico . . . . .	30
8.3	Diagrama do Modelo Conceptual . . . . .	31
<b>9</b>	<b>Propostas de Interface</b>	<b>32</b>
<b>10</b>	<b>Metodologia de Implementação</b>	<b>33</b>
<b>11</b>	<b>Ferramentas utilizadas na implementação</b>	<b>34</b>

---

## INTRODUÇÃO

### 2.1 CONTEXTUALIZAÇÃO

O número de ginásios está a aumentar e estima-se que em 2017 a indústria tenha arrecadado cerca de 73 mil milhões de euros a nível mundial, assumindo, assim, um factor de relevo na economia mundial.

Além do exercício físico, o seguimento nutricional dos seus utilizadores juntamente com um aumento no número de planos de baixo custo faz com que o mercado proveniente de ginásios aumente.

### 2.2 APRESENTAÇÃO DO CASO DE ESTUDO

A **UMFit** é um software de apoio e orientação *Fitness* desenvolvido no âmbito da unidade curricular de Laboratórios de Informática IV. O software arquitectado é ambicioso e, por esta razão, todo o processo que constitui a sua criação foi extremamente rigoroso. Para o desenvolvimento do **UMFit** foram exigidos os melhores padrões e metodologias de trabalho com vista à rotina dos clientes do ginásio ser facilitada, aumentando assim, a satisfação dos mesmos.

Este software tem alguns objectivos muito bem definidos, nomeadamente o de proporcionar aos seus utilizadores experiências de utilização do ginásio mais simples e cómodas, para além de unificar o exercício físico à alimentação mais adequada, consoante os objectivos definidos por cada cliente.

## 2.3 MOTIVAÇÃO E OBJETIVOS

Os ginásios disponibilizam aos seus clientes uma grande variedade de serviços. Tais como, aulas de grupo, serviços de nutrição e planeamento de treinos.

A aplicação permite auxiliar os clientes do ginásio para que estes alcancem os seus objectivos, por exemplo, chegar à sua forma ideal de forma mais rápida e saudável. Para além disso, permite que, de uma forma centralizada, os utilizadores possam ver receitas dos seus planos alimentares. Deste modo, todos os diversos exercícios e planos alimentares do cliente estarão disponíveis através dos seus dispositivos moveis e também em qualquer outra plataforma, por um navegador Internet.

A apresentação ao Utilizador das aulas de grupo pretendidas, com o respectivo número de participantes, é apenas umas das muitas funcionalidades que a aplicação apresenta. Depois de entrar pela primeira vez na aplicação, o utilizador tem a oportunidade de definir uma configuração inicial para o sistema, escolhendo, por exemplo, dentro de uma das aulas de grupo disponíveis, uma que pretenda frequentar. Este poderia também ao longo do seu percurso agendar avaliações consoante a disponibilidade dos instrutores[Funcionalidade Extra].

## 2.4 ESTRUTURA DO RELATÓRIO

Ainda por finalizar...

---

## LEVANTAMENTO DE REQUISITOS

A fase de especificação, consistiu no levantamento de requisitos para o desenvolvimento do sistema da **UMFit**. Este processo teve por base a experiência própria de elementos do grupo em ginásios e na utilização de aplicações semelhantes. Foi também feita uma análise a várias aplicações de ginásios e de *Fitness*, para conseguir distinguir as funcionalidades essenciais que distinguem a **UMFit** das plataformas da mesma área de negócio. Algumas aplicações analisadas foram, "MyFitnessPal", "Academia Fitness BVFamalicão", "Gym WP", entre outras.

### 3.1 REQUISITOS FUNCIONAIS

#### 3.1.1 AUTENTICAÇÃO DE UTILIZADORES

##### Definição de requisitos de utilizador

1. O utilizador deverá introduzir o seu e-mail e a respetiva password para iniciar sessão na aplicação.

##### Especificação de requisitos de sistema

- 1.1. O sistema deve verificar a validade da tentativa de autenticação, verificando se o par e-mail e password inseridos correspondem a algum utilizador existente no sistema. Caso não correspondam, o sistema não deve permitir a autenticação.



### 3.1.2 CRIAÇÃO PLANOS DE TREINO E ALIMENTARES

#### Definição de requisitos de utilizador

1. O instrutor deve inserir as opções "Criar Plano Alimentar" ou "Criar Plano de Treino" e inserir o e-mail do respetivo cliente para atribuir o respetivo plano.

#### Especificação de requisitos de sistema

- 1.1. O sistema deve verificar a validade do e-mail de cliente escolhido e adicionar o plano a esse utilizador. Caso o e-mail de cliente será impossível adicionar o plano ao mesmo.

### 3.1.3 CONSULTAR PLANOS DE TREINO E ALIMENTARES

#### Definição de requisitos de utilizador

1. O cliente deve inserir as opções "Consultar Plano alimentar" ou "Consultar Plano de Treino" de modo a visualizar o respetivo plano.

#### Especificação de requisitos de sistema

- 1.1. O Sistema apresenta os dados do plano alimentar ou de treino. Caso o cliente não tenha um plano este não é mostrado.

### 3.1.4 CRIAÇÃO DE METAS

#### Definição de requisitos de utilizador

1. First item.

#### Especificação de requisitos de sistema

- 1.1. First subitem.

### 3.1.5 MARCAR PRESENÇA NUM ESPAÇO[EXTRA]

#### Definição de requisitos de utilizador

1. First item.

Especificação de requisitos de sistema

1.1. First subitem.

3.1.6 APRESENTAÇÃO DE LOTAÇÃO DOS DIFERENTES ESPAÇOS DO GINÁSIO[EXTRA]

Definição de requisitos de utilizador

1. First item.

Especificação de requisitos de sistema

1.1. First subitem.

3.1.7 CONSULTAR ESTATÍSTICAS DE LOTAÇÃO POR HORÁRIO[EXTRA]

Definição de requisitos de utilizador

1. First item.

Especificação de requisitos de sistema

1.1. First subitem.

### 3.2 REQUISITOS NÃO FUNCIONAIS

1. Um utilizador pode ser Premium ou Standard.
2. Os Premium têm acesso a todas as funcionalidades do ginásio referidas acima.
3. Os utilizadores Standard podem consultar e participar nas aulas de grupo, bem como usufruir das zonas de "cardio", musculação e piscina do ginásio. Contudo, estes últimos não têm acesso a planos de treino e/ou alimentares mas podem, no entanto, visualizar a sua última avaliação e evolução.
4. A última avaliação de um cliente deve apresentar a data e todas as medidas de perímetro e composição corporal, incluindo um gráfico da evolução, adequando a interface à medida de risco associada à saúde do cliente;
5. Os planos de treino e as aulas de grupo devem apresentar a data/hora e um conjunto de imagens ilustrativas;
6. A aplicação deverá ser de fácil uso, com um layout o mais simples possível.
7. A aplicação deverá estar disponível durante os 7 dias da semana, 24 horas por dia.
8. O tempo de resposta da aplicação deve ser o mais curto possível para não influenciar de forma negativa a experiência do utilizador na aplicação.
9. O sistema deve ser suportado por todos os browsers e telemóveis.

## MODELO DE DOMÍNIO

Tendo por base os requisitos previamente recolhidos e analisados, foi desenvolvido o modelo de domínio que visa recolher e representar de maneira geral os conceitos, comportamentos e algumas interações que estarão incluídas no sistema a desenvolver.

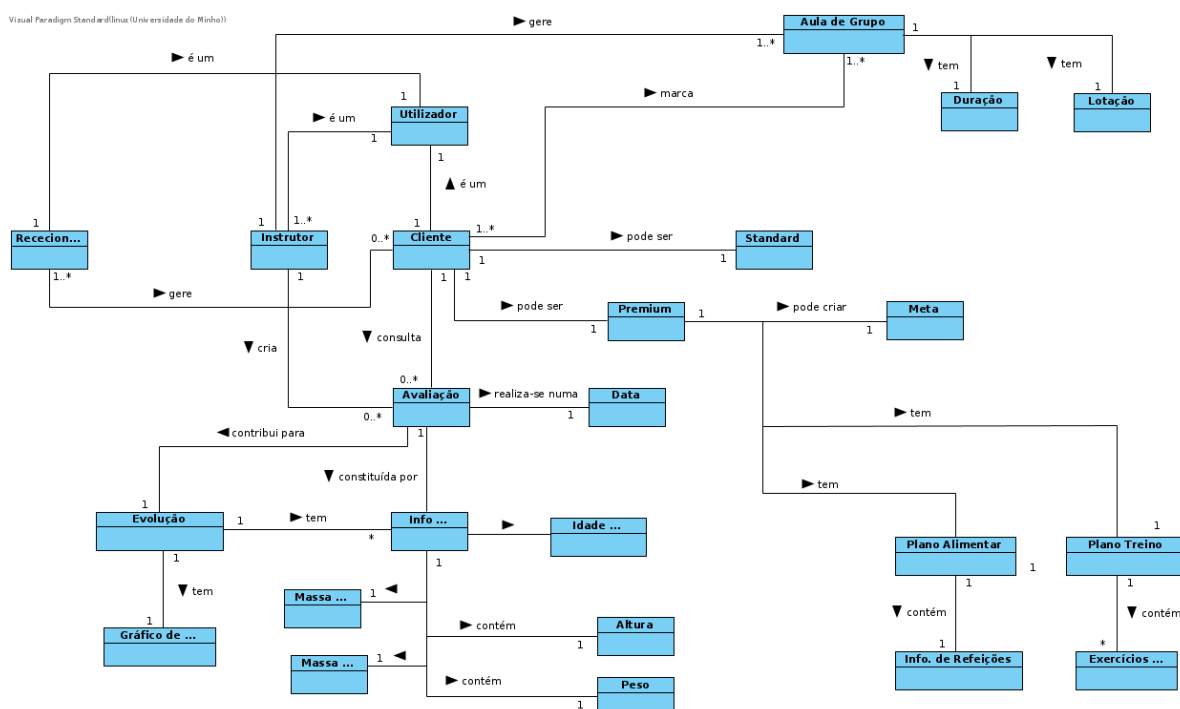


Figura 1: Modelo de Domínio

Numa pequena análise rápida é bastante fácil e intuitivo de perceber que já estão modelados todos os principais objetivos da aplicação nomeadamente a interação Utilizador - Aula com destaque obviamente para as aulas e ainda, também a título de exemplo, alguns dos aspetos que constituem um plano de treino, sejam eles a sua área de treino, nível de dificuldade,

as zonas musculares que a compõe e até as instruções para a sua execução.

## DIAGRAMA DE USE CASES

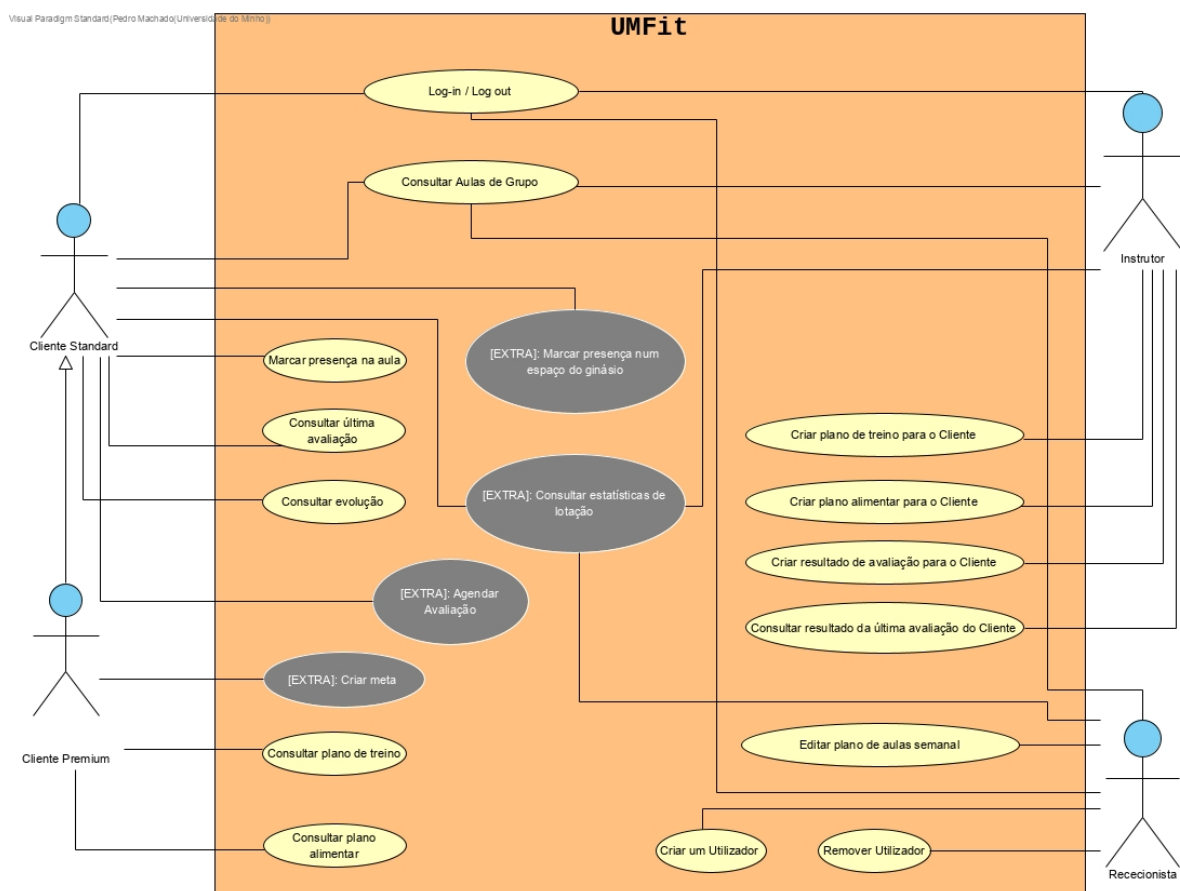


Figura 2: Diagrama de Use Case

---

## ESPECIFICAÇÃO DE USE CASES

### 6.1 CONSULTAR AULAS DE GRUPO

Descrição: Utilizador verifica a informação referente a uma aula de grupo.

Pré-Condição: O Utilizador tem de estar autenticado no sistema e existirem aulas de grupo no sistema.

Pós-Condição: O utilizador vê a informação referente à aula de grupo.

.....

Fluxo Normal:

1. O utilizador seleciona a opção "consultar aulas de grupo".
2. O sistema apresenta a lista com as aulas de grupo e as suas informações respetivas.

### 6.2 CONSULTAR EVOLUÇÃO

Descrição: Utilizador verifica dados da sua evolução.

Pré-Condição: O utilizador ter pelo menos uma avaliação.

Pós-Condição: O utilizador vê os dados da sua evolução.

.....

Fluxo Normal:

1. O Utilizador seleciona a opção "Consultar Evolução";
2. O Sistema apresenta os dados da sua evolução.

### 6.3 CONSULTAR PLANO ALIMENTAR

Descrição: Utilizador verifica o seu plano alimentar.

Pré-Condição: O utilizador ter-lhe atribuído um plano alimentar.

Pós-Condição: O utilizador vê os dados do seu plano alimentar.

.....

Fluxo Normal: 1. O Utilizador seleciona a opção "Consultar Plano alimentar";  
2. O Sistema apresenta os dados do plano alimentar.

### 6.4 CONSULTAR PLANO DE TREINO

Descrição: Utilizador verifica o seu plano de treino.

Pré-Condição: O utilizador ter-lhe atribuído um plano de treino.

Pós-Condição: O utilizador vê os dados do seu plano de treino.



.....

Fluxo Normal:

1. O Utilizador seleciona a opção "Consultar Plano de treino";
2. O Sistema apresenta os dados do plano de treino.

## 6.5 CONSULTAR RESULTADO DA ÚLTIMA AVALIAÇÃO DE UM DADO UTILIZADOR

Descrição: O instrutor consulta a última avaliação.

Pré-Condição: O instrutor está autenticado no sistema.

Pós-Condição: O instrutor obtém os dados da ultima avaliação.

.....

Fluxo Normal:

1. O instrutor seleciona a opção "Consultar a última avaliação";
2. O instrutor insere o email do cliente do qual pretende ver a última avaliação;
3. O sistema valida o cliente escolhido;
4. O sistema apresenta os dados da última avaliação.

Fluxo Exceção 1: [O email inserido não é válido] (passo 3)

- 3.1 O Sistema invalida o email inserido.
- 3.2 O Sistema termina o processo.

Fluxo Exceção 2: [O cliente não tem avaliações] (passo 4)

4.1 O Sistema termina o processo.

## 6.6 CONSULTAR RESULTADO DA ÚLTIMA AVALIAÇÃO

Descrição: O Rececionista verifica dados da sua última avaliação do utilizador escolhido.

Pré-Condição: O Utilizador ter pelo menos uma avaliação e o Rececionista estar autenticado.

Pós-Condição: O Rececionista vê os dados da avaliação.

.....

Fluxo Normal:

1. O Rececionista seleciona a opção "Consultar última Avaliação";
2. O Rececionista insere o e-mail do Utilizador pretendido;
3. O Sistema apresenta a última avaliação do utilizador ao Rececionista.

## 6.7 CONSULTAR ÚLTIMA AVALIAÇÃO

Descrição: Utilizador verifica dados da sua última avaliação.

Pré-Condição: O utilizador ter pelo menos uma avaliação.

Pós-Condição: O utilizador vê os dados da avaliação.

.....

Fluxo Normal:

1. O Utilizador selecciona a opção "Consultar última Avaliação";
2. O Sistema apresenta a última avaliação do utilizador.

## 6.8 CRIAR PLANO ALIMENTAR PARA CLIENTE

Descrição: O Instrutor atribui a um utilizador um plano alimentar.

Pré-Condição: O instrutor está autenticado no sistema.

Pós-Condição: O sistema insere um novo plano alimentar no cliente.

.....

Fluxo Normal:

1. O instrutor selecciona a opção "Criar Plano Alimentar";
2. O instrutor insere o email do cliente a quem deseja atribuir o plano alimentar;
3. O sistema valida o cliente escolhido;
4. O instrutor insere os dados do plano alimentar;
5. O sistema adiciona o plano alimentar ao utilizador.

Fluxo Exceção 1: [O email inserido não é válido] (passo 3)

- 3.1 O Sistema invalida o email inserido.
- 3.2 O Sistema termina o processo.

## 6.9 CRIAR PLANO DE TREINO PARA CLIENTE

Descrição: O Instrutor atribui a um utilizador um plano de treino.

Pré-Condição: O instrutor está autenticado no sistema.

Pós-Condição: O sistema insere um novo plano de treino no cliente.

..... Fluxo Normal:

1. O instrutor seleciona a opção "Criar Plano de treino";
2. O instrutor insere o email do cliente a quem deseja atribuir o plano de treino;
3. O sistema valida o cliente escolhido;
4. O instrutor insere os dados do plano de treino;
5. O sistema adiciona o plano de treino ao utilizador.

Fluxo Exceção 1: [O email inserido não é válido] (passo 3)

- 3.1 O Sistema invalida o email inserido.
- 3.2 O Sistema termina o processo.

## 6.10 CRIAR RESULTADO DE AVALIAÇÃO PARA CLIENTE

Descrição: O Instrutor atribui a um utilizador uma avaliação.

Pré-Condição: O instrutor está autenticado no sistema.

Pós-Condição: O sistema insere uma nova avaliação no cliente.

.....

Fluxo Normal:

1. O instrutor seleciona a opção "Criar Avaliação";
2. O instrutor insere o email do cliente a quem deseja atribuir a avaliação;
3. O sistema valida o cliente escolhido;
4. O instrutor insere os dados da avaliação;
5. O sistema adiciona a avaliação ao utilizador.

Fluxo Exceção 1: [O email inserido não é válido] (passo 3)

- 3.1 O Sistema invalida o email inserido.
- 3.2 O Sistema termina o processo.

## 6.11 CRIAR UTILIZADOR

Descrição: Um novo utilizador é criado no sistema pelo rececionista.

Cenários: Um Emanuel quer entrar no ginásio e a Maria rececionista cria a sua conta.

Pré-Condição: O rececionista tem de estar autenticado no sistema.

Pós-Condição: Um novo utilizador é adicionado ao sistema.

.....

Fluxo Normal: O rececionista insere o nome, e-mail e morada do novo utilizador, juntamente com o seu género e outras informações pertinentes ao registo. O Sistema valida os dados e regista um novo utilizador com credenciais aleatórias;

### Fluxo Excepção:

- 2.1 Os dados fornecidos ao Administrador são inválidos;
- 2.2 O Administrador não cria a nova conta;

## 6.12 EDITAR PLANOS DE AULAS SEMANAL

Descrição: A rececionista altera as aulas de grupo.

Pré-Condição: O rececionista está autenticado no sistema.

Pós-Condição: No sistema são alteradas as informações das aulas de grupo.

.....

### Fluxo Normal:

1. O rececionista seleciona a opção "Editar plano de aulas";
2. O sistema apresenta as opções de "edição de aulas" ou "adição de aulas".
3. O rececionista seleciona a opção edição de aulas.
4. O sistema apresenta uma lista das aulas de grupo existentes.
5. O rececionista escolhe uma aula de grupo para ser editada.
6. O sistema apresenta os dados referentes a essa aula.
7. O rececionista altera os dados da aula.
8. O sistema altera a informação da aula de grupo com os novos dados.

Fluxo Alternativo 1: [O rececionista seleciona adição de aulas] (passo 3)

- 3.1 O sistema pergunta ao rececionista as informações sobre a nova aula.
- 3.2 O rececionista insere as informações referentes a essa aula.
- 3.3 A nova aula de grupo é adicionada ao sistema.

Fluxo Exceção 1: [Não existem aulas disponiveis] (passo 4)

- 4.1 O sistema informa que não existem aulas de grupo no sistema.
- 4.2 O sistema termina o processo.

## 6.13 LOG IN

Descrição: Autenticar um cliente, instrutor ou rececionista.

Pré-Condição: O Utilizador tem de estar registado no Sistema e não pode haver outro Utilizador no Sistema.

Pós-Condição: O Utilizador fica autenticado no Sistema como cliente, instrutor ou rececionista.

.....

Fluxo Normal:

1. O Sistema apresenta as opções de Autenticação;
2. O Utilizador escolhe a opções de entrar como cliente, instrutor ou rececionista e insere as credenciais;
3. O Sistema valida as credenciais inseridas;
4. O Utilizador autentica-se no Sistema;

.....

Fluxo de exceção 1: [As credenciais são inválidas] (passo 3)

- 3.1 O Sistema avisa o Utilizador que as credenciais são inválidas;
- 3.2 O Utilizador sai do Sistema.

#### 6.14 LOG OUT

Descrição: O utilizador faz log out da aplicação.

Pré-Condição: O Utilizador tem de estar autenticado no Sistema.

Pós-Condição: O utilizador saiu do sistema.

.....

Fluxo Normal:

1. O utente selecciona a opção "Log out";
2. O sistema faz log-out do utilizador.

#### 6.15 MARCAR PRESENÇA NA AULA

Descrição: Utilizador escolhe uma aula na qual pretende marcar presença.

Pré-Condição: Utilizador estar autenticado no sistema e existirem vagas para a aula seleccionada.

Pós-Condição: O utilizador fica inscrito na aula escolhida.



.....

Fluxo Normal:

1. O Utilizador selecciona a opção "Consultar Aulas";
2. O Sistema apresenta a lista aulas.
3. O Utilizador selecciona a aula.
4. O Sistema inscreve o utilizador na aula.

## 6.16 REMOVER UTILIZADOR

Descrição: Um Utilizador é removido do sistema pelo Rececionista.

Cenários: Para facilitar a gestão de Utilizador no Sistema, o Rececionista tem a possibilidade de remover Utilizadores;

Pré-Condição: O Rececionista tem de estar autenticado no Sistema.

Pós-Condição: Um Utilizador é removido do Sistema.

.....

Fluxo Normal:

1. O Rececionista escolhe a opção de "Remover Utilizador";
2. O Rececionista escolhe o e-mail do Utilizador que pretende remover;
3. O Rececionista escolhe a opção de "Remover do Sistema";
4. O Rececionista selecciona o Utilizador que quer remover;
5. O Sistema remove o Utilizador e todo o seu conteúdo;

---

## ARQUITETURA DA CAMADA DE NEGÓCIOS

### 7.1 CAMADA DE NEGÓCIOS

Como consequência lógica da análise conjunta dos requisitos e do modelo de domínio apresentado, identificamos e caracterizamos as classes e desenvolvemos uma pequena descrição da arquitetura da solução a desenvolver. Esta descrição foi essencial na construção do diagrama de classes representativo da arquitetura da camada de negócios.

### 7.2 DICIONÁRIO DAS PRINCIPAIS CLASSES

**Utilizador (Utilizador)** - Corresponde à representação no sistema dos utilizadores da aplicação e contém a informação pessoal, informação de acesso e nível de dificuldade.

**Avaliação (Avaliação)** - Representa uma componente necessária a uma ou várias receitas e após confeccionada é integrada na mistura que a determinada receita específica. Esta classe guarda a informação referente à data, massa gorda, massa marga, imc, id, peso, idade metabólica, altura, se foi realizada ou não (foi\_realizada), nif do cliente, e nif do instrutor.

**Aula Grupo (Aula\_Grupo)** - É a classe que representa a aula de grupo a ser escolhida pelo cliente. Esta classe contém um id, uma dificuldade, uma data, uma lotação atual, um nome, uma lotação máxima, e o nif do instrutor.

**Plano Treino (Plano\_Treino)** - Classe que representa uma cadeia de instruções que termina com a confecção de um dado prato. Contém um conjunto de exercícios, uma data de início, uma data de fim, e um id.

**Plano Alimentar (Plano\_Alimentar)** - Classe que representa o plano alimentar e as respectivas refeições. Contém um conjunto de refeições, um nome, um id, uma data de início, uma data de fim, uma frequência e um número de refeições livres.

**Espaço Ginásio (Espaço\_Ginasio)** - Classe que representa a ocupação do ginásio. Contém um nome, e uma lotação.

### 7.3 DESCRIÇÃO DA ARQUITETURA

Os clientes têm um plano de treinos que contém vários exercícios, plano alimentar que contém refeições adicionalmente, um conjunto de aulas de grupo e também de as avaliações já realizadas.

Os planos de treino vão ter exercícios realizados pelos clientes, e os planos alimentares vão ter um conjunto de refeições que o cliente pode consumir. As refeições contém o nome, o id, e a sua descrição e os exercícios contém o nome do mesmo, um id, e o número de repetições pretendidas.

## 7.4 DIAGRAMA DE CLASSES

O modelo de classes derivado através da descrição apresentada é abaixo. Adicionalmente, às entidades mencionadas foi também criado um facade. A criação deste teve como objetivo, mediar a interação com toda a camada de negócios para que a solução apresentada fosse modular e composicional com um sistema congruente com uma arquitetura de três camadas.

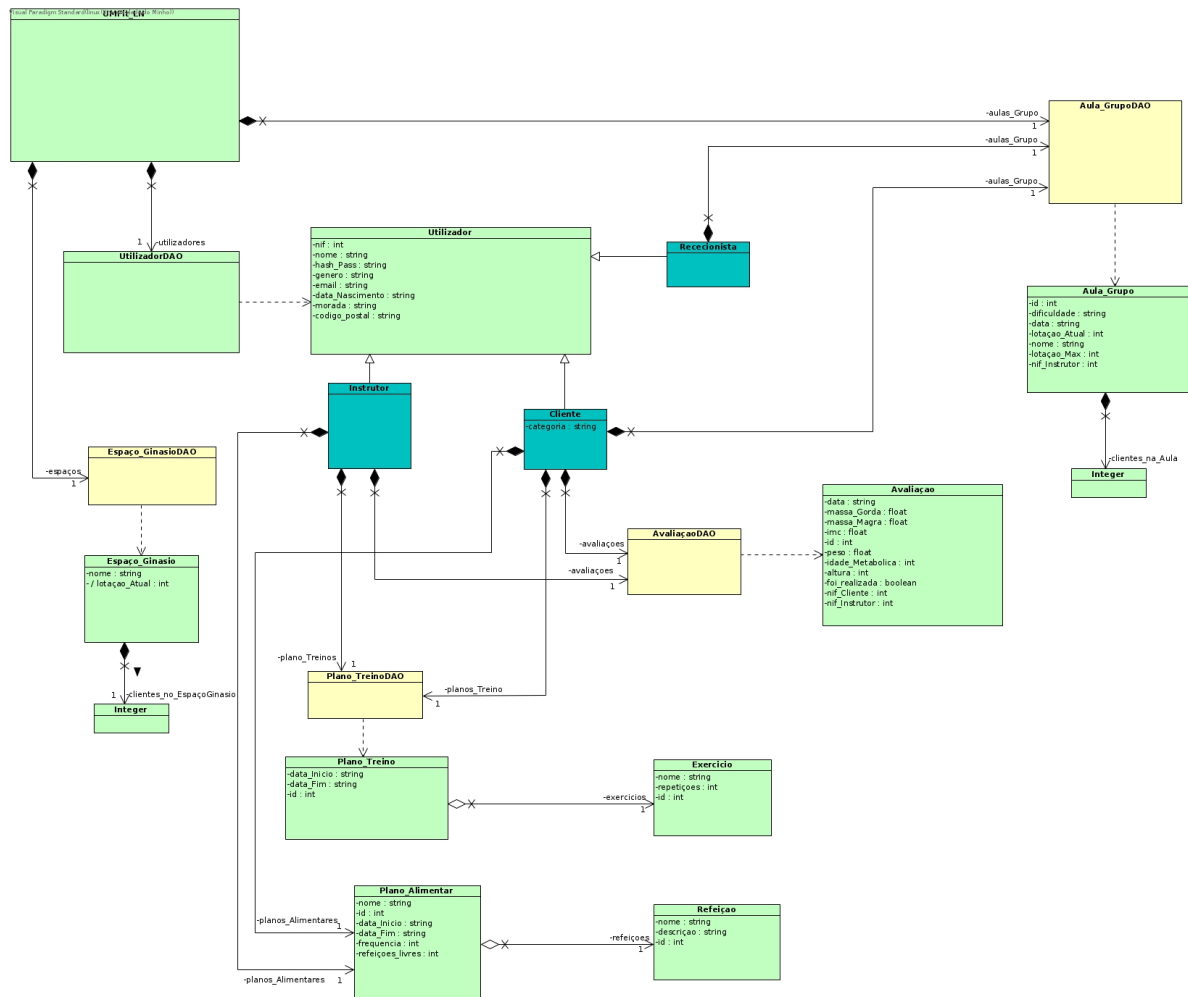


Figura 3: Diagrama de Classes

---

## CAMADA DE DADOS

A camada de dados, sendo que suportará a natureza distribuída da solução de software proposta, terá que garantir a consistência, integridade, segurança e acessibilidade aos dados guardados. Por esta razão, decidimos desenvolver uma base de dados num Sistema de Gestão de Base de Dados Relacional.

### 8.1 MODELO LÓGICO

O modelo lógico apresentado, resulta essencialmente, da estruturação da informação importante a guardar da modelação da camada de negócios apresentada.

Para cada uma das principais classes do modelo de classes, mais precisamente, as classes que representam a Avaliação, Aula de Grupo, Utilizador, Plano de Treino, Plano Alimentar, Espaço no Ginásio. Para representar o relacionamento entre estas Relações foram associadas as seguintes tabelas, **Clientes\_na\_AulaGrupo**, **Cliente\_no\_EspaçoGinasio**, **Plano-Treino\_do\_Cliente**, **Exercicio\_do\_Plano\_Treino**, **Plano\_Alimentar\_Refeicoes**, **Plano-Alimentar\_do\_Cliente**.

## 8.2 DIAGRAMA DO MODELO LÓGICO

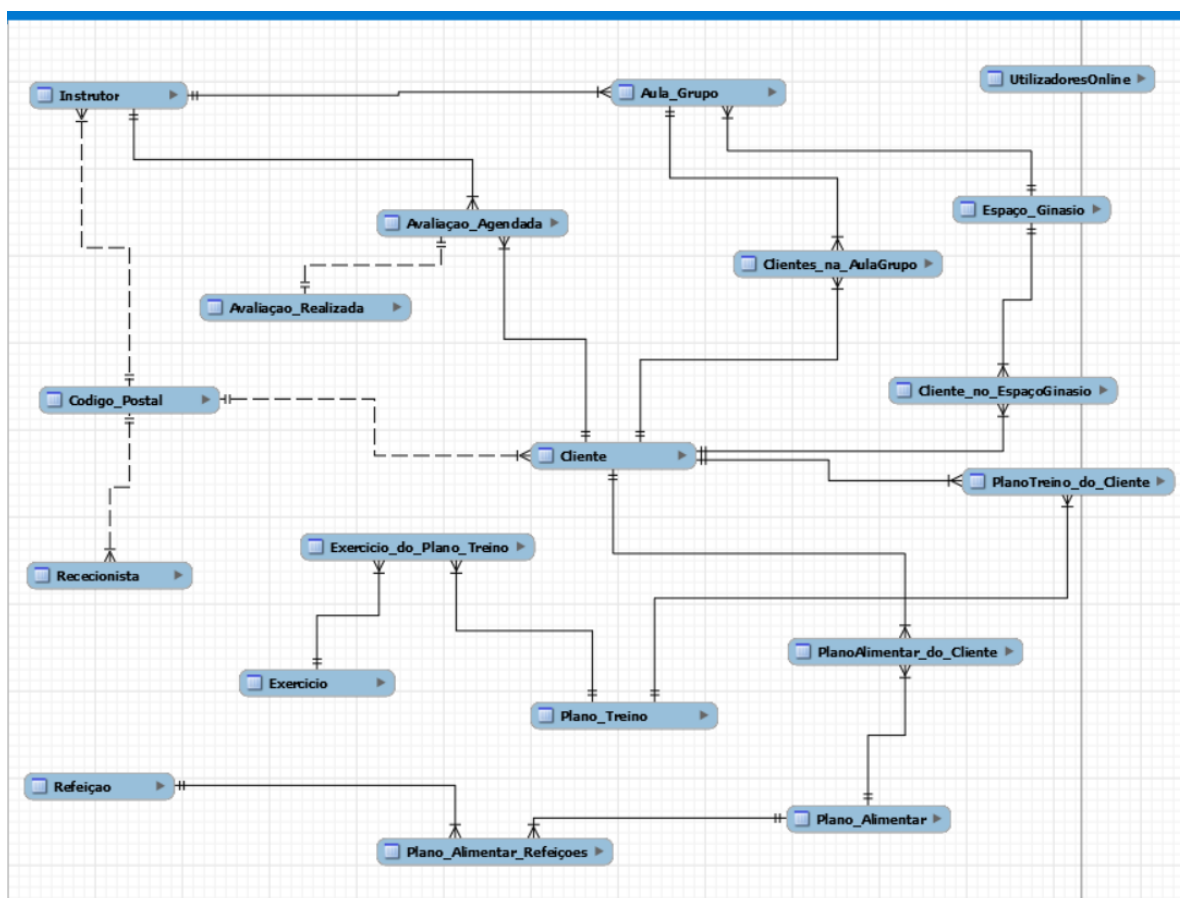


Figura 4: Diagrama do Modelo Lógico

### 8.3 DIAGRAMA DO MODELO CONCEPTUAL

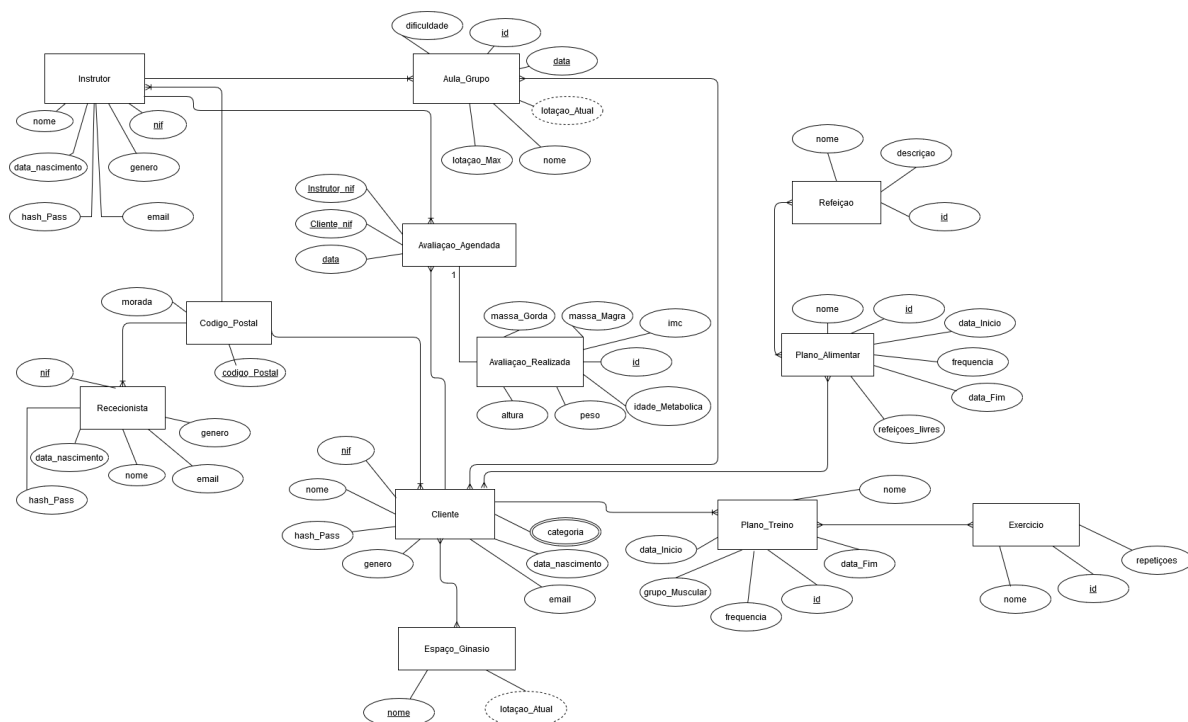


Figura 5: Diagrama do Modelo Conceptual

---

## PROPOSTAS DE INTERFACE



# 10

---

## METODOLOGIA DE IMPLEMENTAÇÃO

---

## FERRAMENTAS UTILIZADAS NA IMPLEMENTAÇÃO

A principal ferramenta usada na implementação da solução de software apresentada foi as frameworks de desenvolvimento ASP.NET Core e Entity Framework. Esta ferramenta mostrou ser fundamental pois, para além de estar extremamente bem documentada, permitiu maximizar o tempo dos programadores, dado que a grande maioria das funcionalidades de suporte a aplicações multiplataforma e de recurso a object relational mapping (ORM), que consistem no mapeamento dos diferentes objetos em entidades numa base de dados, encontram-se implementadas à partida.

O gestor de base de dados escolhido foi o MySQL. Este usa o modelo relacional e integra-se transparentemente com a hierarquia de ferramentas e funcionalidades do ASP.NET core.

Adicionalmente, para reduzir o número de inconformidades associadas ao uso de diferentes ambientes de desenvolvimento todos os programadores usaram o IDE Rider da JetBrains. Por fim, para de melhor forma compreender e testar o meio de interação entre os futuros utilizadores e o produto final, no decorrer do desenvolvimento, foram usados browsers de internet para avaliar a interface visual deste.