

# UNIVERSIDADE DO MINHO

DESENVOLVIMENTO DE SISTEMAS DE SOFTWARE

DEZEMBRO DE 2019

MEDIA CENTER

---

DIOGO RIBEIRO, a84442  
LUÍS MAIA, a84442  
RUI REIS, a84930



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>4</b>
<b>2</b>	<b>Modelação</b>	<b>5</b>
2.1	Modelo de Domínio . . . . .	5
2.2	Especificação dos Use Case . . . . .	5
2.2.1	Fazer Login . . . . .	6
2.2.2	Iniciar sessão como convidado . . . . .	6
2.2.3	Fazer Logout . . . . .	7
2.2.4	Fazer Upload . . . . .	7
2.2.5	Fazer Download . . . . .	7
2.2.6	Criar Conta . . . . .	8
2.2.7	Remover Conta . . . . .	8
2.2.8	Editar Categoria . . . . .	8
2.2.9	Criar Playlist . . . . .	9
2.2.10	Remover Playlist . . . . .	9
2.2.11	Editar playlist . . . . .	10
2.2.12	Reproduzir Conteúdo . . . . .	10
2.2.13	Adicionar Conteúdo à Playlist . . . . .	11
2.2.14	Remover Conteúdo da Playlist . . . . .	11
2.2.15	Adicionar Álbum à Playlist . . . . .	11
2.2.16	Apagar Conteúdo . . . . .	12
2.2.17	Editar Utilizador . . . . .	12
2.2.18	Alterar Password . . . . .	12
2.2.19	Fazer convite de amigo . . . . .	13
2.2.20	Responder a convite de amigo . . . . .	13
2.2.21	Remover amigo . . . . .	14
2.3	Modelo de Use Cases . . . . .	14
2.4	Diagrama de Classe . . . . .	16
2.4.1	MediaCenter . . . . .	16
2.4.2	MediaPlayer . . . . .	17
2.5	Diagrama de Sequência . . . . .	18
2.5.1	Fazer Login . . . . .	18
2.5.2	Iniciar sessão como convidado . . . . .	19
2.5.3	Fazer Logout . . . . .	19
2.5.4	Fazer Upload . . . . .	20
2.5.5	Fazer Download . . . . .	21
2.5.6	Criar Conta . . . . .	21
2.5.7	Remover Conta . . . . .	22
2.5.8	Editar Categoria . . . . .	23
2.5.9	Criar Playlist . . . . .	24
2.5.10	Remover Playlist . . . . .	24
2.5.11	Editar playlist . . . . .	25
2.5.12	Reproduzir Conteúdo . . . . .	26
2.5.13	Adicionar Conteúdo à Playlist . . . . .	27
2.5.14	Remover Conteúdo da Playlist . . . . .	27
2.5.15	Adicionar Álbum à Playlist . . . . .	28
2.5.16	Apagar Conteúdo . . . . .	28
2.5.17	Editar Utilizador . . . . .	29
2.5.18	Alterar Password . . . . .	29
2.5.19	Fazer convite de amigo . . . . .	30
2.5.20	Responder a convite de amigo . . . . .	30
2.5.21	Remover amigo . . . . .	31
2.6	Diagrama de Packages . . . . .	31
2.7	Protótipo da Interface . . . . .	32
2.8	Diagrama de Estado . . . . .	39
2.9	Diagrama de Componentes . . . . .	39
2.10	Diagrama de Atividade . . . . .	40
2.11	ORM - Object-Relational Mapping . . . . .	40

<b>3</b>	<b>Implementação</b>	<b>41</b>
3.1	JDBC – Java DataBase Connectivity . . . . .	41
3.2	Tabelas do Modelo Relacional . . . . .	42
3.3	Detalhes Relevantes da Implementação . . . . .	42
3.3.1	Implementação Inicial . . . . .	42
3.3.2	Utilização do VLC . . . . .	42
3.3.3	Funcionalidade em Rede . . . . .	43
3.3.4	Reprodução de Conteúdo . . . . .	43
3.4	Descrição da Interface . . . . .	43
<b>4</b>	<b>Conclusões</b>	<b>47</b>

## **Lista de Figuras**

1	Diagrama do Modelo de Domínio.	5
2	Diagrama de Use Cases Geral.	14
3	Diagrama de Use Cases Geral de Gestão de Contas.	14
4	Diagrama de Use Cases Geral de Gestão de Playlists.	15
5	Diagrama de Use Cases de Gestão de Conteúdo.	15
6	Diagrama de Use Cases de Gestão de Amigos.	15
7	Diagrama de Classe do MediaCenter.	17
8	Diagrama de Classe do MediaPlayer.	17
9	Diagrama de Sequência de Fazer Login.	18
10	Diagrama de Sequência de Iniciar sessão como convidado.	19
11	Diagrama de Sequência de Fazer Logout.	19
12	Diagrama de Sequência de Fazer Upload.	20
13	Diagrama de Sequência de Fazer Download.	21
14	Diagrama de Sequência de Criar Conta.	21
15	Diagrama de Sequência de Remover Conta.	22
16	Diagrama de Sequência de Editar Categoria.	23
17	Diagrama de Sequência de Criar Playlist.	24
18	Diagrama de Sequência de Remover Playlist.	24
19	Diagrama de Sequência de Editar Playlist.	25
20	Diagrama de Sequência de Reproduzir Conteúdo.	26
21	Diagrama de Sequência de Adicionar Conteúdo à Playlist.	27
22	Diagrama de Sequência de Remover Conteúdo da Playlist.	27
23	Diagrama de Sequência de Adicionar Álbum à Playlist.	28
24	Diagrama de Sequência de Apagar Conteúdo.	28
25	Diagrama de Sequência de Editar Utilizador.	29
26	Diagrama de Sequência de Alterar Password.	29
27	Diagrama de Sequência de Fazer convite de amigo.	30
28	Diagrama de Sequência de Responder a convite de amigo.	30
29	Diagrama de Sequência de Remover amigo.	31
30	Diagrama de Packages do Sistema.	31
31	Interface de Login.	32
32	Interface de Utilizador.	32
33	Interface de Convidado.	33
34	Interface de Alteração de Password.	33
35	Interface de Reset de Password.	33
36	Interface de Upload de Conteúdo.	34
37	Interface de Criação de Nova Playlist.	34
38	Interface de Gestão da Playlist.	35
39	Menu de edição de uma playlist.	36
40	Interface de Gestão de Categorias do Conteúdo.	36
41	Interface de Alteração de Categoria.	37
42	Interface de Administração.	37
43	Interface de Criação de Conta.	37
44	Interface de Erro Básico na Criação de Conta.	38
45	Interface de Informação Adicional Sobre uma Conta.	38
46	Menu de gestão de amigos.	38
47	Diagrama de Estado para o login.	39
48	Diagrama de Componentes do Sistema.	39
49	Diagrama de Atividade do Use Case fazer upload.	40
50	Diagrama ER.	42
51	Formulário de Login.	44
52	Vista principal de um cadastrado.	45
53	Formulário de Upload.	45
54	Alterar Categoria de um Conteúdo.	46

# 1 Introdução

Para o trabalho da cadeira de Desenvolvimento de Sistemas de Software foi-nos apresentado uma proposta de um "Media Center" para partilha de video/audio, modelando este de acordo com os cenários apresentados.

O relatório tem como objetivo proporcionar ao leitor uma percepção clara da funcionalidade do projeto, ao ponto de daqui a 5 anos, entendermos o que esta faz e o porquê das decisões feitas.

Este é composto pela modelação do projeto, dividido em vários modelos e diagramas feitos ao longo das 3 fases do projeto, que nos permitiram, no final, uma fácil passagem para código, visto que a estrutura do projeto está feita na modelação.

De forma introdutória apresentamos o modelo de domínio, bem como uma selecção de *use cases* base. Que foram utilizados no sentido do desenvolvimento e crescente adição de camadas de complexidade a este sistema.

Nos capítulos seguintes, apresentamos as *mockups* iniciais, utilizadas para definir o *GUI* a ser utilizado, bem como decisões de *design* na implementação deste.

Que permitiram finalmente, obter um sistema funcional que cumpre todos os requisitos pedidos pelos docentes, e mais, todos numa versão *beta*. Para além disso, foi possível introduzir os conceitos de funcionamento em rede, persistência de dados e reprodução de conteúdo.

## 2 Modelação

### 2.1 Modelo de Domínio

Como um primeiro rascunho do funcionamento e relações que o nosso sistema deveria incorporar, e após leitura do enunciado proposto, conseguimos obter o modelo de domínio a seguir apresentado, que visa representar o relacionamento entre as entidades do sistema por nós identificadas.

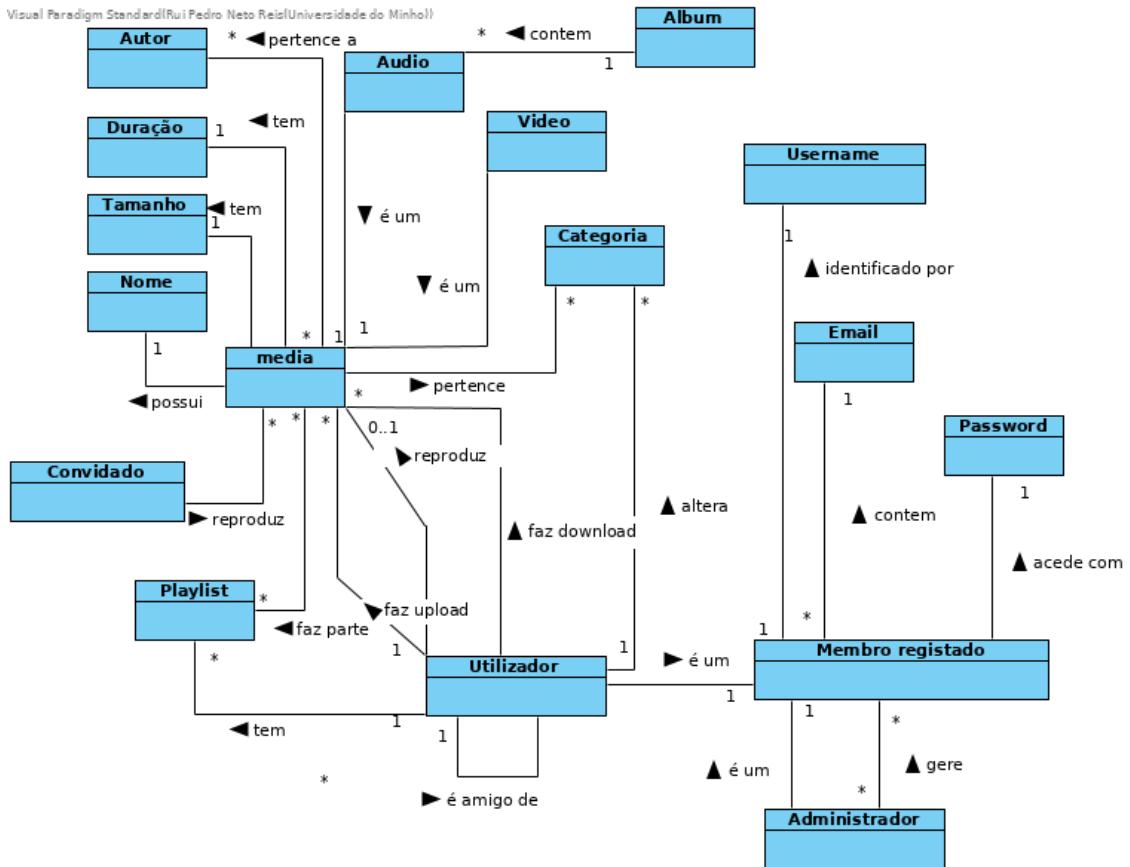


Figura 1: Diagrama do Modelo de Domínio.

### 2.2 Especificação dos Use Case

Após reuniões com os nossos consultores e análise de diferentes estatísticas de potenciais clientes, foi possível obter uma lista dos seguintes *use cases* que o nosso sistema deve definir de forma a poder obter alguma quota do mercado. *Use cases* que especificaremos de seguida.

1. Fazer Login.
2. Iniciar sessão como convidado.
3. Fazer Logout.
4. Fazer Upload.
5. Fazer Download.
6. Criar Conta.
7. Remover Conta.
8. Editar Categoria.

9. Criar Playlist.
10. Remover Playlist.
11. Editar playlist.
12. Reproduzir Conteúdo.
13. Adicionar Conteúdo à Playlist.
14. Remover Conteúdo da Playlist.
15. Adicionar Álbum à Playlist.
16. Apagar Conteúdo.
17. Editar Utilizador.
18. Alterar Password.
19. Fazer convite de amigo.
20. Responder a convite de amigo.
21. Remover amigo.

Que resulta na seguinte especificação de descrição mais detalhada dos *use cases*.

### **2.2.1 Fazer Login**

- **Use Case:** Fazer login.

**Descrição:** Iniciar sessão como membro registado.

**Cenários:** O Manuel está registado no sistema e pretende iniciar sessão.

**Pré-condição:** Não possuir sessão iniciada, estar registado no sistema.

**Pós-condição:** Sessão iniciada como Utilizador.

**Fluxo Normal:**

1. Utilizador introduz username e password.
2. Sistema valida acesso.

**Fluxo de Alternativo 1: [Utilizador esqueceu-se da password](passo 1)**

- 2.1 Utilizador fornece email e username
- 2.2 Sistema envia email com nova password e atualiza a mesma.

**Fluxo de Exceção 1: [Username não existe](passo 2)**

- 2.1 Sistema avisa que o username não existe.

### **2.2.2 Iniciar sessão como convidado**

- **Use Case:** Iniciar sessão como convidado.

**Descrição:** Inicia uma sessão como convidado.

**Cenários:** O Rui, que não está registado no sistema, pretende apenas ouvir música e, assim sendo, inicia sessão como convidado.

**Pré-condição:** Não possuir sessão iniciada.

**Pós-condição:** Sessão iniciada como convidado.

**Fluxo Normal:**

1. Utilizador indica que quer fazer login como convidado.
2. Sistema valida o acesso como convidado

### **2.2.3 Fazer Logout**

- **Use Case:** Fazer logout

**Descrição:** Fazer logout

**Cenários:** O Manuel já não quer ouvir mais música e pretende terminar a sua sessão no sistema.

**Pré-condição:** Sessão iniciada.

**Pós-condição:** Não possuir sessão iniciada.

**Fluxo Normal:**

1. Utilizador indica que quer fazer logout.
2. Sistema termina a sessão.

### **2.2.4 Fazer Upload**

- **Use Case:** Fazer upload.

**Descrição:** Fazer upload de um ficheiro.

**Cenários:** O Manuel pretende fazer upload da sua música favorita para o sistema para que todos os colegas possam ouvir.

**Pré-condição:** Sessão iniciada como utilizador.

**Pós-condição:** Ficheiro disponível na plataforma.

**Fluxo Normal:**

1. Utilizador indica que quer fazer upload.
2. Utilizador fornece ficheiro.
3. Sistema verifica se o ficheiro é válido.
4. Utilizador fornece os dados relativos ao ficheiro.
5. Sistema avisa que upload foi feito com sucesso.

**Fluxo de Exceção 1: [Tipo de ficheiro inválido](passo 3)**

- 3.1 Sistema alerta que o tipo de ficheiro selecionado é inválido.

### **2.2.5 Fazer Download**

- **Use Case:** Fazer download

**Descrição:** Descarregar um determinado ficheiro da plataforma

**Cenários:** O Manuel pretende descarregar uma música que está no sistema.

**Pré-condição:** Sessão iniciada como Utilizador.

**Pós-condição:** É criada uma cópia local do ficheiro na máquina do utilizador.

**Fluxo Normal:**

1. Utilizador indica que pretende descarregar o ficheiro.
2. Sistema pede para utilizador indicar directória onde pretende fazer download.
3. Utilizador indica qual a diretoria pretendida.
4. Sistema descarrega o ficheiro para a diretoria indicada.

**Fluxo Exceção 1:[Diretoria inválida](passo 3)**

- 3.1.1 Sistema alerta que a diretoria indicada não é válida.

**Fluxo Exceção 2:[Utilizador não tem permissões para descarregar](passo 1)**

- 1.1.1 Sistema alerta que o utilizador não tem permissões para descarregar o ficheiro.

## 2.2.6 Criar Conta

- **Use Case:** Criar conta.

**Descrição:** Criar conta.

**Cenários:** A Paula, que tem permissões de administrador, cria uma conta para o Manuel.

**Pré-condição:** Sessão Iniciada como administrador.

**Pós-condição:** Nova conta criada.

**Fluxo Normal:**

1. Administrador indica que pretende criar uma nova conta.
2. Administrador indica o tipo de conta que pretende criar.
3. Administrador indica o email, username e password.
4. Sistema valida username, password e email.
5. Sistema cria a conta e retorna mensagem de sucesso.

**Fluxo de Exceção 1: [Username já existente](passo 4)**

- 6.1 Sistema indica que username já existe.

## 2.2.7 Remover Conta

- **Use Case:** Remove uma conta do mediaCenter.

**Descrição:** Remove a conta de um membro registado.

**Cenários:** O Ricardo vai sair do apartamento e a Paula quer eliminar a conta dele.

**Pré-condição:** Sessão iniciada como administrador, assume que o cadastrado que se pretende remover existe.

**Pós-condição:** Conta removida da plataforma.

**Fluxo Normal:**

1. Administrador indica qual o username do membro registado que pretende remover.
2. Sistema apaga os ficheiros que pertencem ao username indicado.
3. Sistema remove utilizador do plataforma.
4. Sistema retorna mensagem de sucesso.

**Fluxo Alternativo 1:[Ficheiro possui mais donos](passo 2)**

- 2.1 Sistema decrementa o número de donos do conteúdo.

- 2.2 Regressa ao passo 3.

## 2.2.8 Editar Categoria

- **Use Case:** Editar a categoria de um conteúdo.

**Descrição:** Editar a categoria de um conteúdo.

**Cenários:** O Ricardo pretende editar uma categoria de um conteúdo.

**Pré-condição:** Sessão iniciada como utilizador, conteúdo existente.

**Pós-condição:** Categoria do conteúdo editada.

**Fluxo Normal:**

1. Utilizador quer editar a categoria do conteúdo.
2. Utilizador indica o conteúdo a ser afetado.
3. Sistema apresenta as categorias atuais do conteúdo.
4. Utilizador escolhe adicionar uma categoria.
5. Utilizador indica a nova categoria a adicionar.
6. Sistema adiciona a categoria do conteúdo indicado.

#### **Fluxo Alternativo 1:[Escolhe alterar uma categoria](passo 4)**

- 4.1 Utilizador seleciona a categoria a alterar.
- 4.2 Utilizador indica a nova categoria.
- 4.3 Sistema altera a categoria selecionada do conteúdo indicado.

#### **Fluxo Alternativo 2:[Escolhe eliminar uma categoria](passo 4)**

- 4.1 Utilizador seleciona a categoria a remover.
- 4.2 Sistema remove a categoria selecionada do conteúdo indicado.

### **2.2.9 Criar Playlist**

- **Use Case:** Cria uma playlist

**Descrição:** Cria uma playlist associada ao utilizador.

**Cenários:** O Manuel pretende criar uma playlist com as suas músicas favoritas.

**Pré-condição:** Sessão iniciada como utilizador.

**Pós-condição:** Playlist criada.

**Fluxo Normal:**

1. Utilizador indica o nome da playlist.
2. Utilizador indica uma descrição.
3. Sistema pede o tipo de playlist.
4. Utilizador indica o tipo de playlist.
5. Sistema cria a playlist.

#### **Fluxo Alternativo 2:[Criar Playlist Aleatória](passo 4)**

- 4.1 Sistema cria uma playlist aleatória.

#### **Fluxo Alternativo 3:[Criar Playlist por categoria](passo 4)**

- 4.1 Sistema pergunta qual a categoria.
- 4.2 Utilizador indica a categoria.
- 4.3 Sistema cria uma playlist para a categoria.

#### **Fluxo Alternativo 4:[Criar Playlist por Artista](passo 4)**

- 4.1 Sistema pergunta qual o Artista.
- 4.2 Utilizador indica o Artista.
- 4.3 Sistema cria uma playlist para o Artista.

### **2.2.10 Remover Playlist**

- **Use Case:** Remover playlist.

**Descrição:** Remover uma playlist associada à conta.

**Cenários:** O Manuel pretende eliminar uma das suas playlists.

**Pré-condição:** Sessão iniciada como utilizador, playlist escolhida válida.

**Pós-condição:** Playlist removida.

**Fluxo Normal:**

1. Utilizador pede para remover uma playlist.
2. Sistema inicializa o processo de remoção de playlist.
3. Utilizador seleciona uma playlist.
4. Sistema pede confirmação (sim ou não).
5. Utilizador responde que sim.
6. Sistema remove a playlist.

#### **Fluxo Alternativo 1:[Utilizador responde que não](passo 5)**

- 5.1 Sistema não remove a playlist.

### 2.2.11 Editar playlist

- **Use Case:** Editar playlist

**Descrição:** Altera a descrição ou o nome de uma playlist.

**Cenários:** O Manuel quer alterar a descrição da playlist.

**Pré-condição:** Sessão iniciada como utilizador.

**Pós-condição:** Nome ou descrição da playlist alterada.

**Fluxo Normal:**

1. Utilizador indica qual playlist pretende alterar.
2. Utilizador indica novo nome.
3. Sistema altera nome da playlist.
4. Utilizador indica nova descrição.
5. Sistema altera descrição da playlist.
6. Sistema confirma alterações feitas.

**Fluxo Alternativo 1:[Utilizador não indica novo nome](passo 2)**

- 2.1 Sistema não altera o nome.
- 2.2 Regressa ao passo 4.

**Fluxo Alternativo 2:[Utilizador não indica nova descrição](passo 4)**

- 4.1 Sistema não altera a descrição.
- 4.2 Regressa ao passo 6.

### 2.2.12 Reproduzir Conteúdo

- **Use Case:** Reproduzir Conteúdo.

**Descrição:** Conteúdo ser reproduzido.

**Cenários:** O Ricardo pretende reproduzir uma música existente no sistema.

**Pré-condição:** Sessão Iniciada.

**Pós-condição:** Conteúdo escolhido é reproduzido.

**Fluxo Normal:**

1. Utilizador seleciona uma das opções de reproduzir conteúdo.
2. Utilizador escolhe reproduzir um conteúdo.
3. Sistema verifica se há algum conteúdo a ser reproduzido.
4. Sistema reproduz conteúdo.

**Fluxo Alternativo 1:[Escolhe diminuir Volume](passo 2)**

- 2.1 Sistema altera o volume.

**Fluxo Alternativo 2:[Escolhe aumentar Volume](passo 2)**

- 2.1 Sistema altera o volume.

**Fluxo Alternativo 3:[Escolhe ativar modo Shuffle](passo 2)**

- 2.1 Sistema verifica se está em modo shuffle.
- 2.2 Sistema reconhece que modo Shuffle não se encontra ativo.
- 2.3 Sistema ativa o modo shuffle.

**Fluxo Alternativo 4:[Modo shuffle ativo](passo 2.1)**

- 2.1.1 Sistema desativa o modo shuffle.

**Fluxo Alternativo 5:[Escolhe parar Conteúdo](passo 2)**

2.1 Sistema para a reprodução de conteúdo.

**Fluxo Alternativo 6:**[Escolhe avançar Conteúdo](passo 2)

2.1 Sistema avança o item a ser reproduzido.

**Fluxo Alternativo 7:**[Escolhe retroceder Conteúdo](passo 2)

2.1 Sistema regressa ao conteúdo reproduzido anteriormente.

### 2.2.13 Adicionar Conteúdo à Playlist

- **Use Case:** Adicionar Conteúdo à playlist.

**Descrição:** Adiciona conteúdo a uma playlist.

**Cenários:** O Manuel pretende adicionar uma música a uma playlist.

**Pré-condição:**Sessão iniciada como membro registado, playlist existe.

**Pós-condição:**Conteúdo adicionado à playlist.

**Fluxo Normal:**

1. Utilizador indica o conteúdo que quer adicionar a uma playlist.
2. Utilizador indica a playlist a que vai adicionar o conteúdo.
3. Sistema valida se o conteúdo é repetido.
4. Sistema adiciona o conteúdo à playlist.

**Fluxo Alternativo 1:**[Conteúdo já se encontra na playlist](passo 2)

2.1 Sistema alerta que o conteúdo já se encontra na playlist.

### 2.2.14 Remover Conteúdo da Playlist

- **Use Case:** Remover Conteúdo da playlist.

**Descrição:** Remove conteúdo a uma playlist.

**Cenários:** O Ricardo pretende remover um ficheiro de uma das suas playlist.

**Pré-condição:** Sessão iniciada como utilizador, playlist existe e conteúdo existe na playlist.

**Pós-condição:** Playlist sem o conteúdo.

**Fluxo Normal:**

1. Utilizador indica que pretende remover o ficheiro da playlist.
2. Sistema remove o conteúdo da playlist.

### 2.2.15 Adicionar Álbum à Playlist

- **Use Case:** Adicionar um álbum a uma Playlist.

**Descrição:** Adicionar um álbum a uma playlist.

**Cenários:** O Manuel pretende adicionar um álbum novo a uma playlist.

**Pré-condição:** Sessão iniciada como utilizador, playlist existe e album selecionado existe.

**Pós-condição:** Álbum é adicionado à playlist.

**Fluxo Normal:**

1. Utilizador indica um álbum para adicionar.
2. Álbum é adicionado à playlist.

### 2.2.16 Apagar Conteúdo

- **Use Case:** Apagar Conteúdo.

**Descrição:** Apagar um determinado ficheiro do sistema.

**Cenários:** O Manuel pretende apagar uma música do sistema.

**Pré-condição:** Sessão iniciada como utilizador, assumindo que fez upload do ficheiro e que este existe.

**Pós-condição:** Ficheiro é apagado se apenas existir 1 dono, caso contrário, o utilizador é removido da lista de donos.

**Fluxo Normal:**

1. Utilizador indica ficheiro que deseja apagar.
2. Sistema retira utilizador da lista de donos.
3. Sistema retira direito de download ao utilizador.

**Fluxo Alternativo 1:[Utilizador é o único dono](passo 3)**

- 3.1 Sistema remove o conteúdo de todas as playlists do sistema.
- 3.2 Sistema apaga o conteúdo.

### 2.2.17 Editar Utilizador

- **Use Case:** Editar Utilizador

**Descrição:** Editar a informação relativa a um utilizador.

**Cenários:** O Ricardo pretende alterar as informações da sua conta.

**Pré-condição:** Sessão iniciada como membro registado.

**Pós-condição:** Informação da conta atualizada.

**Fluxo Normal:**

1. Utilizador indica que pretende editar a sua conta.
2. Sistema apresenta os dados da conta.
3. Utilizador altera o email.
4. Utilizador indica que pretende guardar as alterações.
5. Sistema efetua as alterações.

**Fluxo Alternativo 1:[Utilizador pretende alterar password](passo 3)**

- 3.1 «include» Alterar Password.
- 3.2 Regressa ao passo 4.

### 2.2.18 Alterar Password

- **Use Case:** Alterar Password

**Descrição:** Alterar a password de uma conta.

**Cenários:** Cenário 1.

**Pré-condição:** Sessão iniciada como cadastrado.

**Pós-condição:** Password alterada.

**Fluxo Normal:**

1. Utilizador indica que pretende alterar a sua password.
2. Sistema pede para introduzir a password antiga.
3. Utilizador introduz password antiga.
4. Sistema pede para introduzir a password nova.
5. Utilizador introduz password nova.
6. Sistema pede para confirmar a password nova.

7. Utilizador reintroduz password nova.
8. Sistema autentica dados inseridos.
9. Sistema altera password do utilizador.

**Fluxo de Exceção 1:[confirmação de nova password errada](passo 8)**

- 8.1.1 Sistema avisa que a confirmação da nova password está errada.

**Fluxo de Exceção 2:[password fraca](passo 8)**

- 8.2.1 Sistema avisa que a nova password introduzida é demasiado fraca.

**Fluxo de Exceção 3:[password antiga errada](passo 8)**

- 8.3.1 Sistema avisa que a password antiga introduzida está errada.

### **2.2.19 Fazer convite de amigo**

- **Use Case:** Fazer convite de amigo.

**Descrição:** Fazer convite de amigo.

**Cenários:** Jorge pretende adicionar Guilherme como seu amigo, pois tem gostos semelhantes.

**Pré-condição:** Sessão iniciada como utilizador.

**Pós-condição:** Convite de amigo feito.

**Fluxo Normal:**

1. Utilizador indica username do utilizador que pretende adicionar como amigo.
2. Sistema elabora e envia um convite de amigo.

**Fluxo de Exceção 1: [Utilizador não existe](passo 1)**

- 1.1.1 Sistema avisa que o utilizador indicado não existe.

**Fluxo de Exceção 2: [Utilizador já é amigo](passo 1)**

- 1.2.1 Sistema avisa que o utilizador indicado já é amigo.

**Fluxo de Exceção 3: [Utilizador já foi convidado](passo 1)**

- 1.3.1 Sistema avisa que o utilizador indicado já foi convidado.

### **2.2.20 Responder a convite de amigo**

- **Use Case:** Responder a convite de amigo.

**Descrição:** Responder a convite de amigo.

**Cenários:** Guilherme recebeu um pedido de amizade no sua inbox e pretende aceitar.

**Pré-condição:** Sessão iniciada como utilizador e assumindo que username indicado é remetente de um convite.

**Pós-condição:** Convite respondido.

**Fluxo Normal:**

1. Utilizador indica username do remetente do convite.
2. Sistema pede se pretende aceitar ou recusar.
3. Utilizador indica que pretende aceitar convite.
4. Sistema adiciona utilizador à lista de amigos.
5. Sistema elimina o convite de amigo.

**Fluxo de Alternativo 1: [Utilizador indica que pretende recusar convite](passo 2)**

- 2.1 Regressa ao passo 5.

### 2.2.21 Remover amigo

- Use Case: Remover amigo.

**Descrição:** Remover amigo da lista de amigos.

**Cenários:** O Jorge e o Pedro chatearam-se e por isso querem deixar de ser amigos.

**Pré-condição:** Sessão iniciada como utilizador, utilizador alvo é amigo.

**Pós-condição:** Amigo removido.

**Fluxo Normal:**

1. Utilizador selecciona amigo que pretende remover.
2. Sistema remove amigo da lista de amigos.

## 2.3 Modelo de Use Cases

Dos *use cases* apresentados é possível obter a seguinte distribuição e organização sobre os diferentes potenciais atores deste sistema.

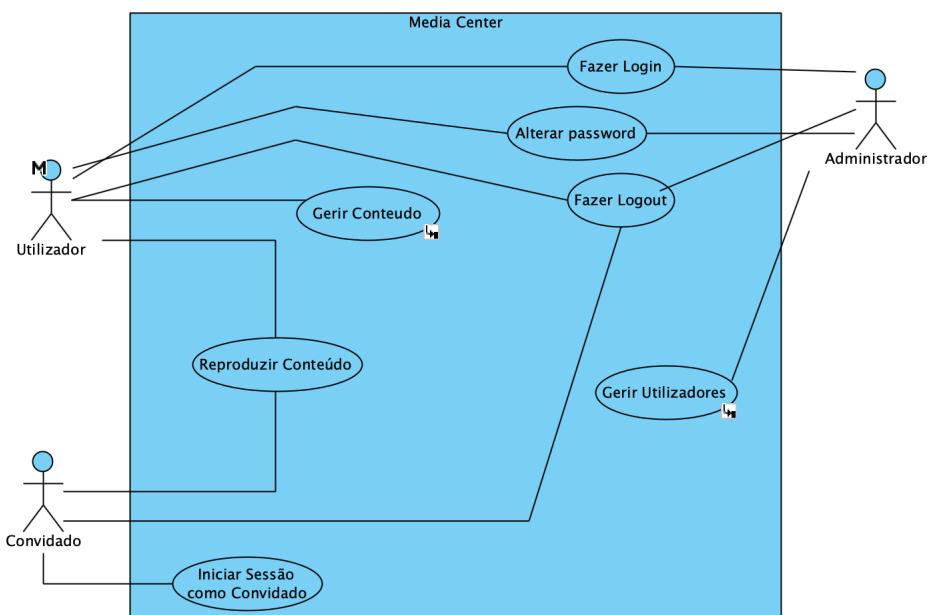


Figura 2: Diagrama de Use Cases Geral.

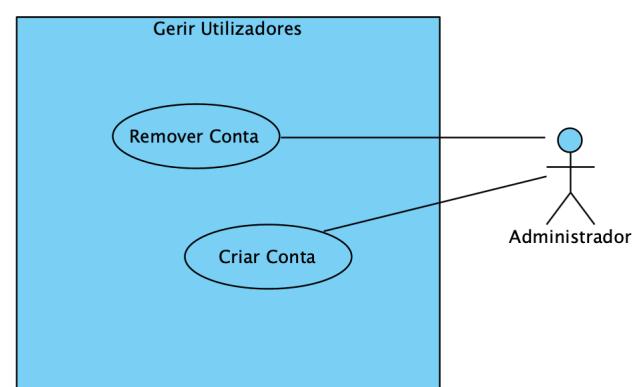


Figura 3: Diagrama de Use Cases Geral de Gestão de Contas.

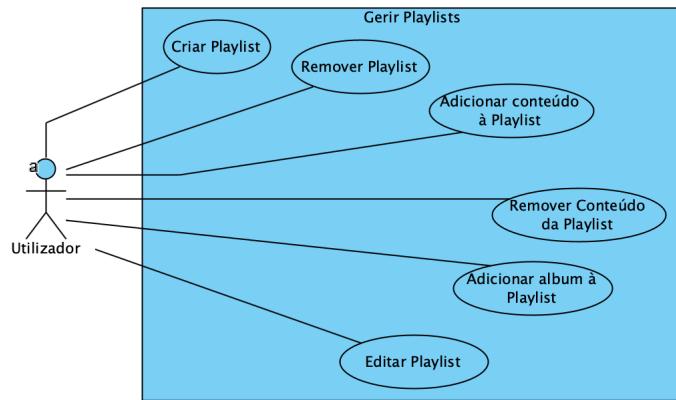


Figura 4: Diagrama de Use Cases Geral de Gestão de Playlists.

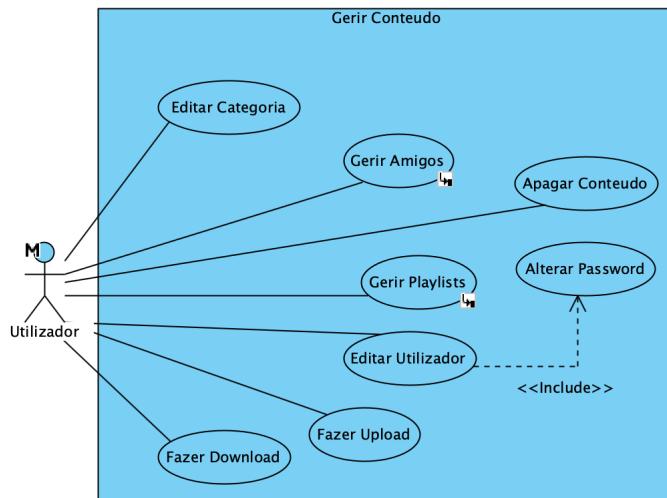


Figura 5: Diagrama de Use Cases de Gestão de Conteúdo.

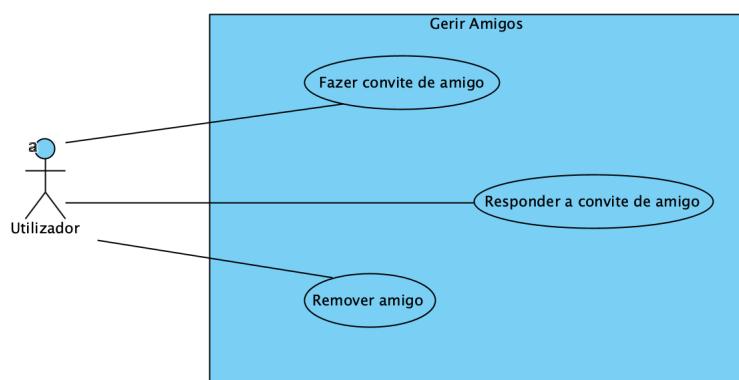


Figura 6: Diagrama de Use Cases de Gestão de Amigos.

## 2.4 Diagrama de Classe

Tendo em conta que o sistema por si só possui definidas duas componentes principais, por si só, chegamos a conclusão de que seria necessária a descrição de dois diagramas de classes distintos. Um deles para o sistema em si e outro para representar a interface que é utilizada na reprodução de conteúdo, que é um processo externo à lógica de negócio.

### 2.4.1 MediaCenter

Ao longo das diferentes versões deste diagrama de classes, certas decisões chave foram tomadas de forma a garantir a eficiência do sistema. Decisões essas que visam garantir que só são obtidos estritamente os recurso necessários da base de dados, visto que o acesso a esta pode ser custoso. Tais decisões são, por exemplo:

- Ao aceder a uma *Playlist*, obtemos da base de dados todos os *Conteudos* desta. Pois, há potencial para todos serem reproduzidos, sendo desnecessário retirar elemento a elemento.
- Ao aceder a um *Utilizador* possuímos apenas acesso aos códigos dos *Conteudos* e *Playlists* deste, pois não é necessário ter acesso direto aos seus *Conteudos*, visto que esta componente é apenas importante para permissões de *download* e remoção, todos podem tocar o conteúdo de todos. Para além disso, não faria sentido obter todas as *Playlists* deste, pois no máximo o utilizador vai reproduzir apenas uma.
- Ao aceder a um *Conteudo* temos acesso direto a todas as *Categorias* e *Donos* deste, pois a qualquer dado momento, e tendo em conta um sistema em rede, há potencial para vários utilizadores a pretender fazer download do conteúdo, o que implica várias verificações de *ownership* do dado ficheiro, o que iria sobrecarregar a base de dados caso tivéssemos de verificar utilizador a utilizador. Para além disso, obtemos todas as categorias de forma a facilitar o sistema de catalogação por categorias.
- Ao aceder a um *Album* temos acesso direto a todos os conteúdos deste, pois se há acesso a um álbum então há potencial para as músicas desse álbum serem acedidas pelo utilizador consecutivamente, o que permite aliviar carga da base de dados.

Que no final permitiu obter o seguinte e final, pelo menos na versão *beta*, diagrama de classes:

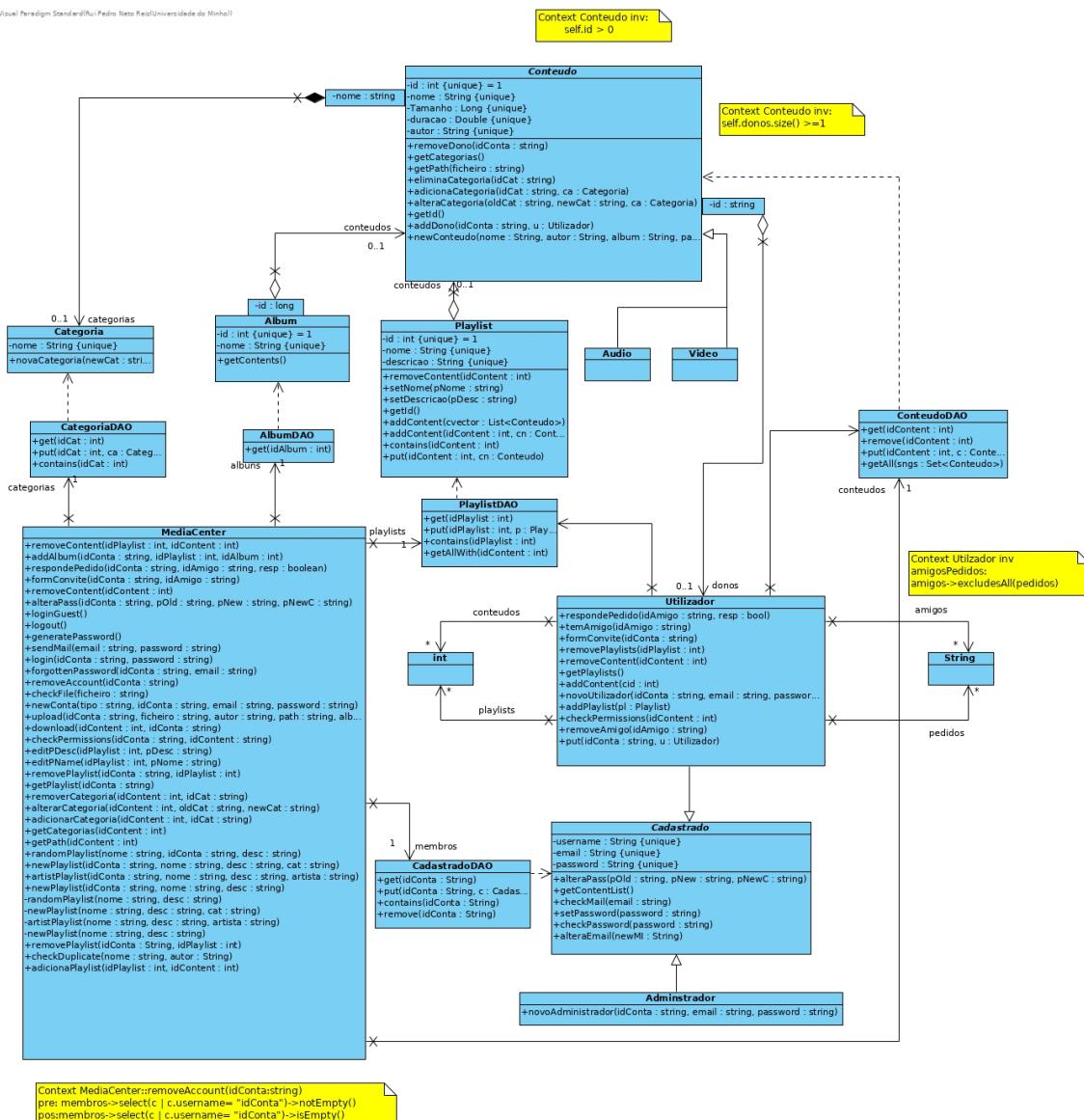


Figura 7: Diagrama de Classe do MediaCenter.

#### 2.4.2 MediaPlayer

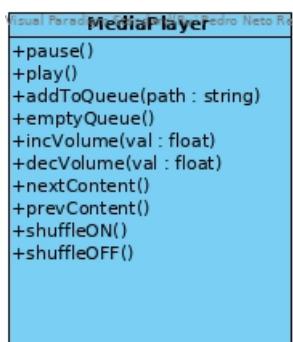


Figura 8: Diagrama de Classe do MediaPlayer.

## 2.5 Diagrama de Sequência

### 2.5.1 Fazer Login

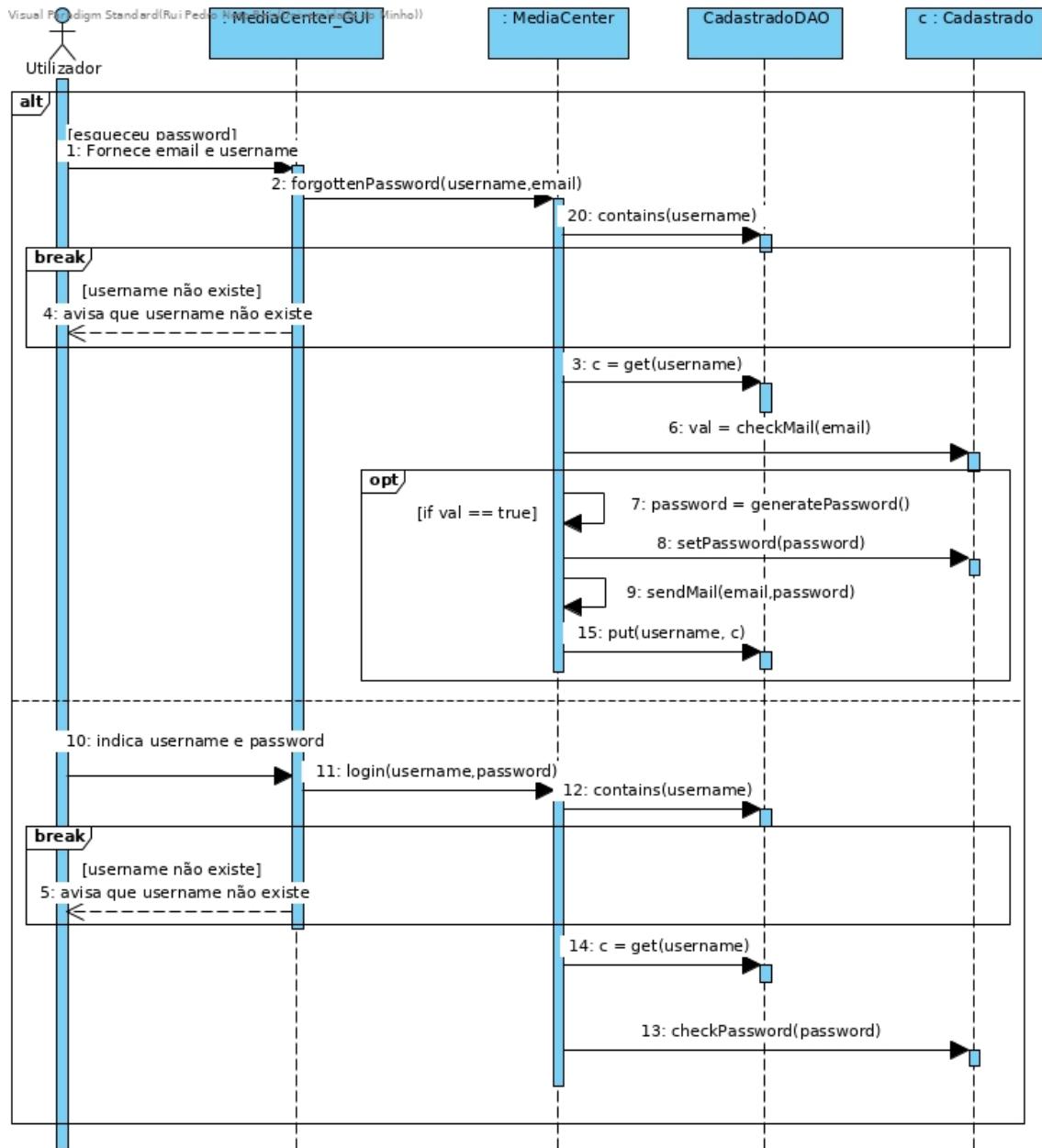


Figura 9: Diagrama de Sequência de Fazer Login.

### 2.5.2 Iniciar sessão como convidado

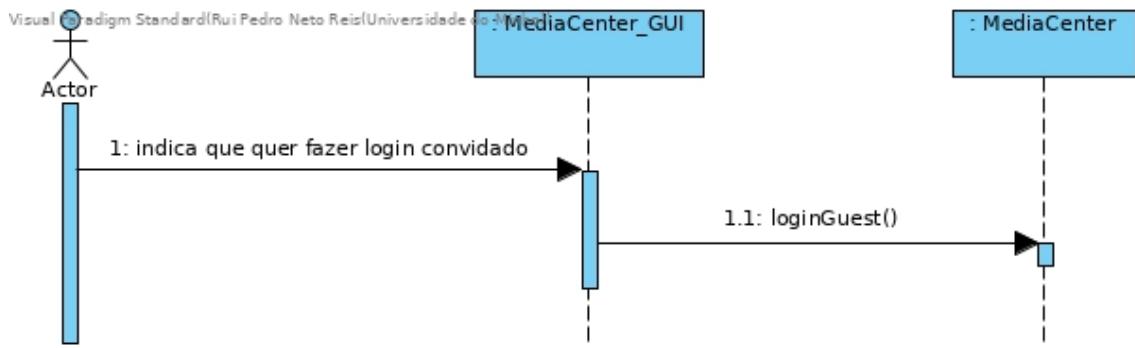


Figura 10: Diagrama de Sequência de Iniciar sessão como convidado.

### 2.5.3 Fazer Logout

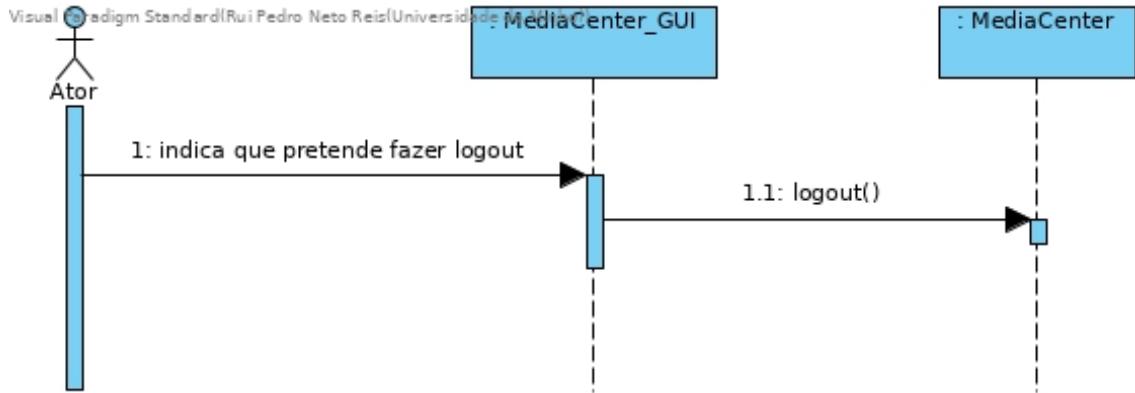


Figura 11: Diagrama de Sequência de Fazer Logout.

#### 2.5.4 Fazer Upload

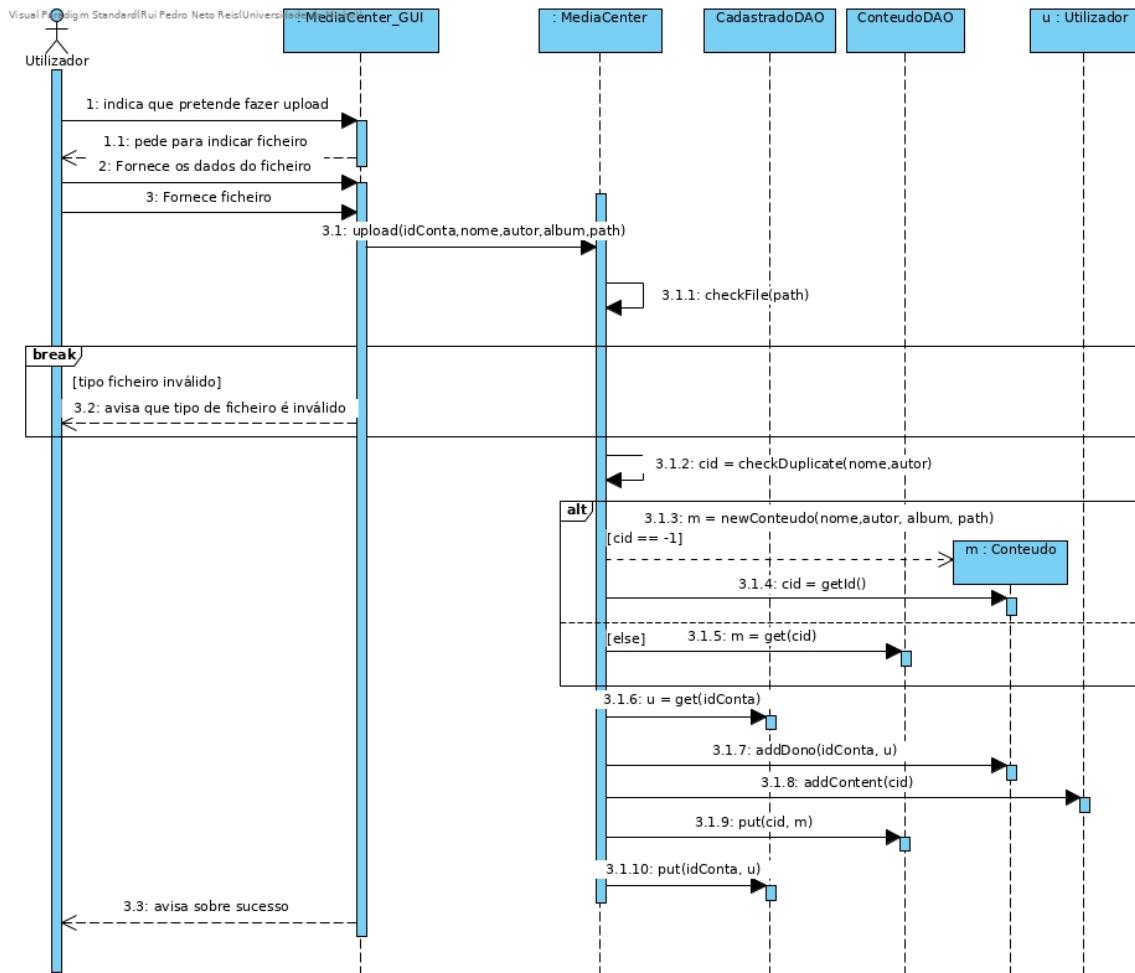


Figura 12: Diagrama de Sequência de Fazer Upload.

### 2.5.5 Fazer Download

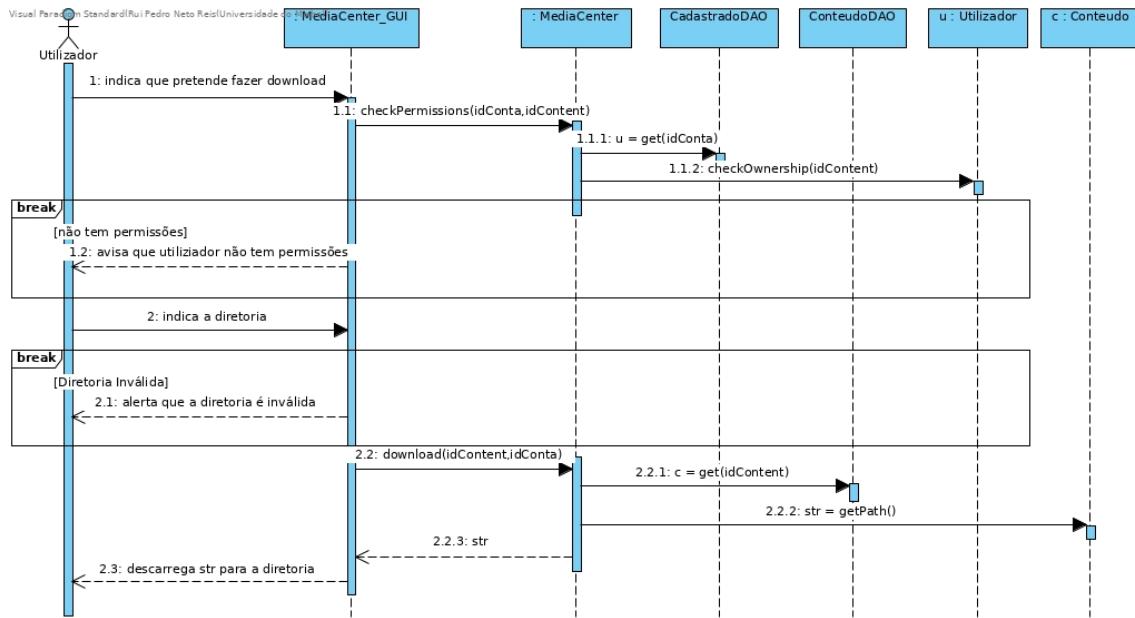


Figura 13: Diagrama de Sequência de Fazer Download.

### 2.5.6 Criar Conta

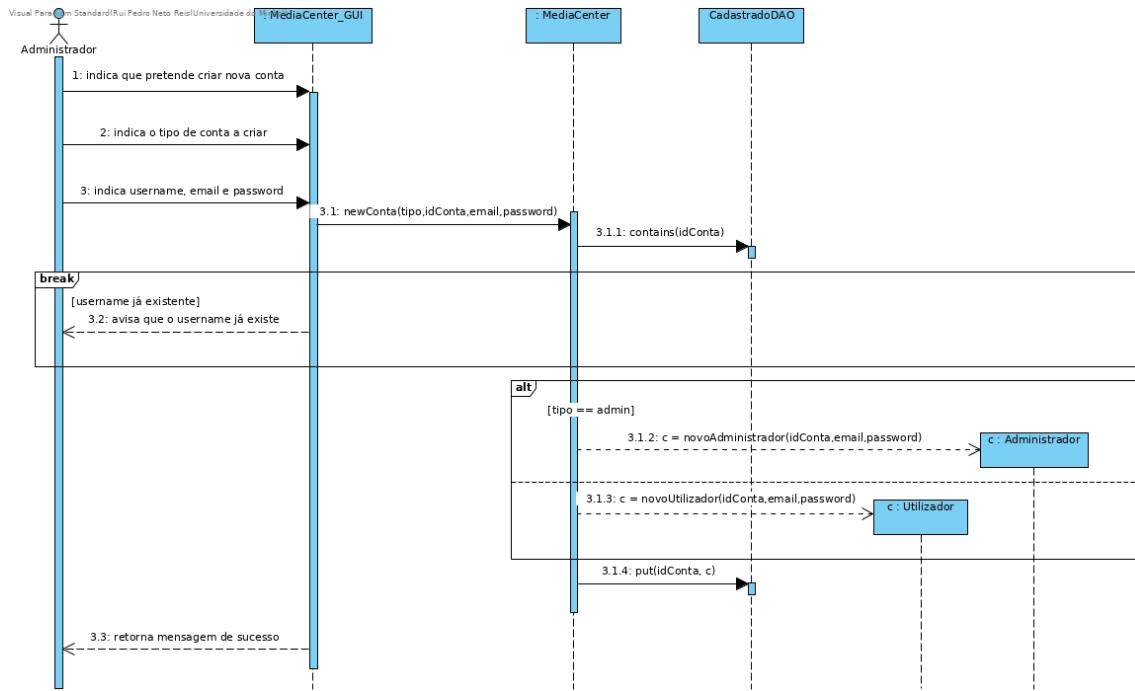


Figura 14: Diagrama de Sequência de Criar Conta.

### 2.5.7 Remover Conta

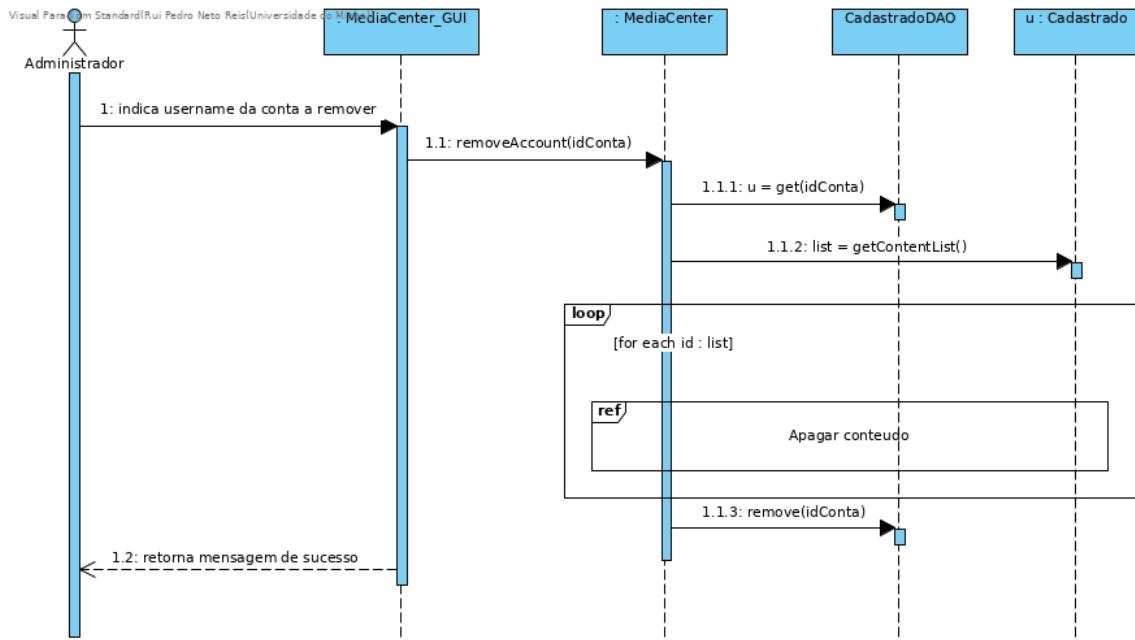


Figura 15: Diagrama de Sequência de Remover Conta.

### 2.5.8 Editar Categoria

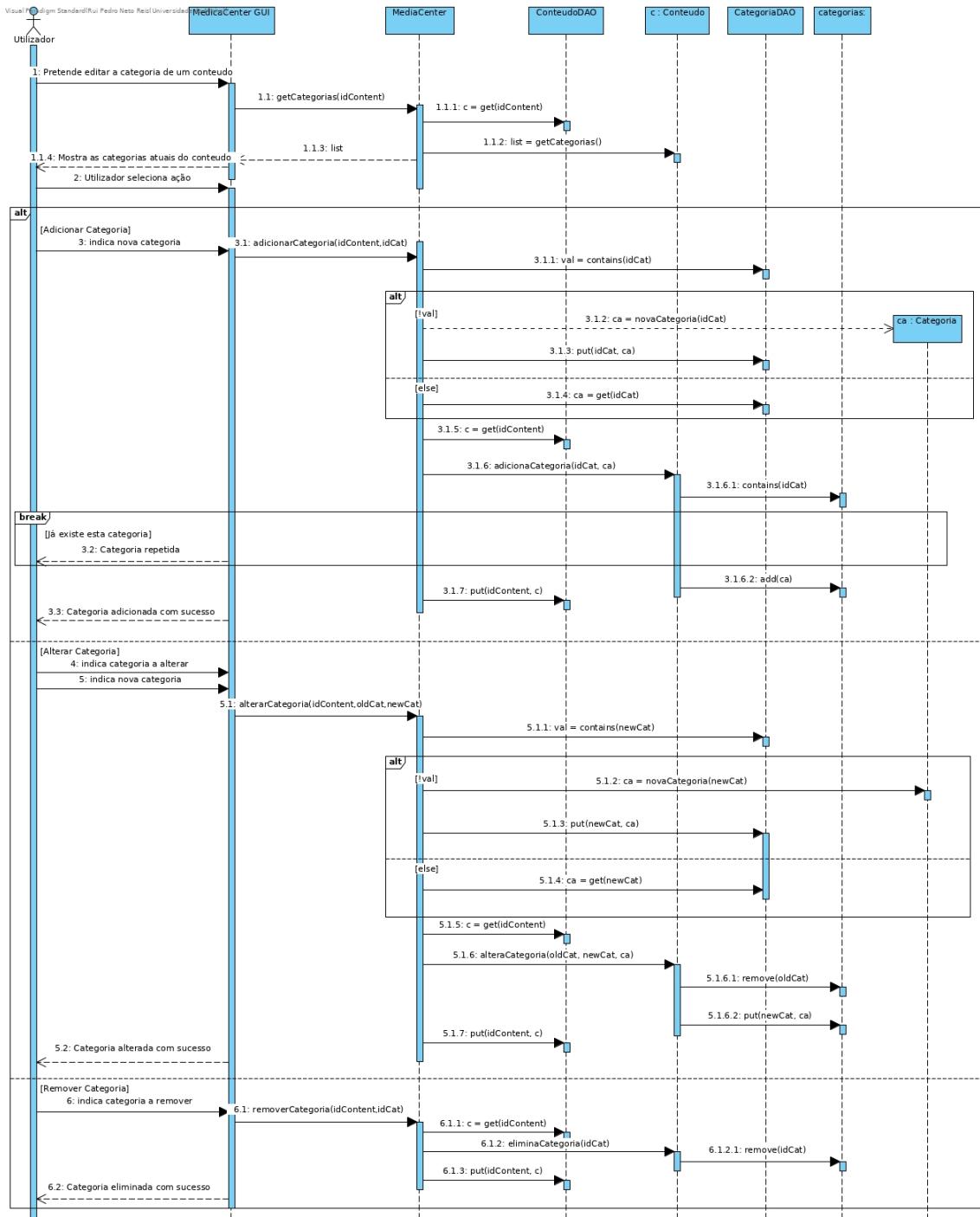


Figura 16: Diagrama de Sequência de Editar Categoria.

### 2.5.9 Criar Playlist

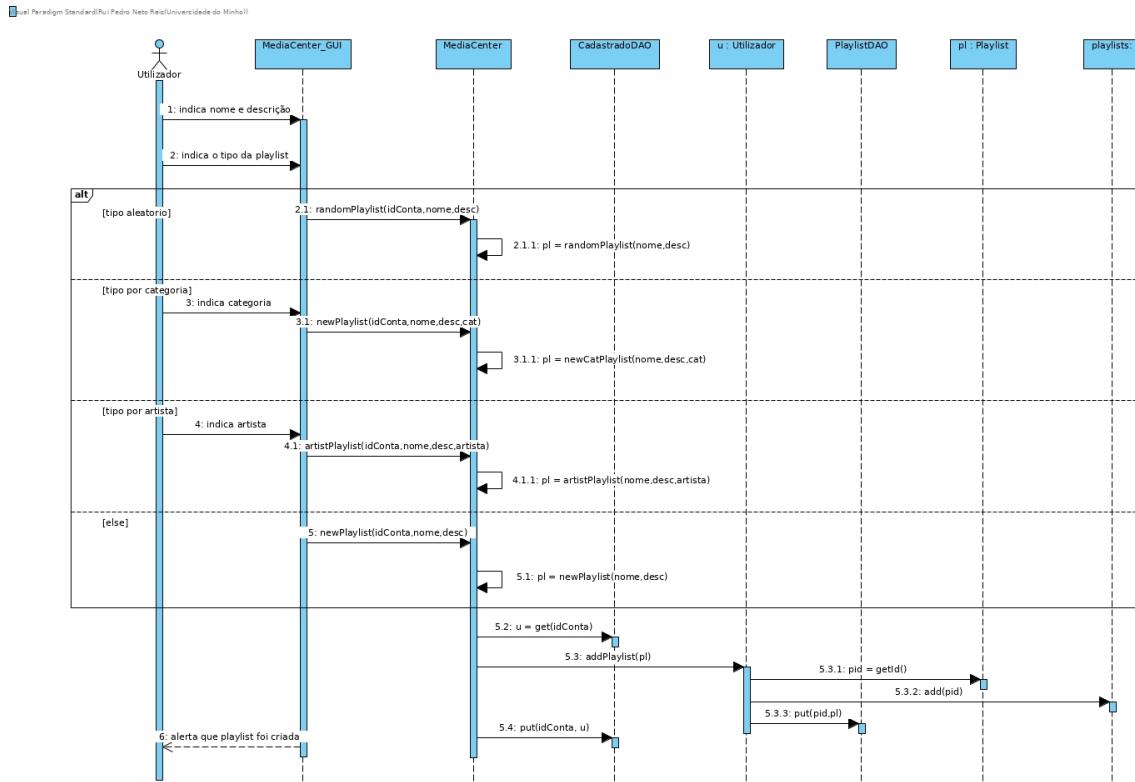


Figura 17: Diagrama de Sequência de Criar Playlist.

### 2.5.10 Remover Playlist

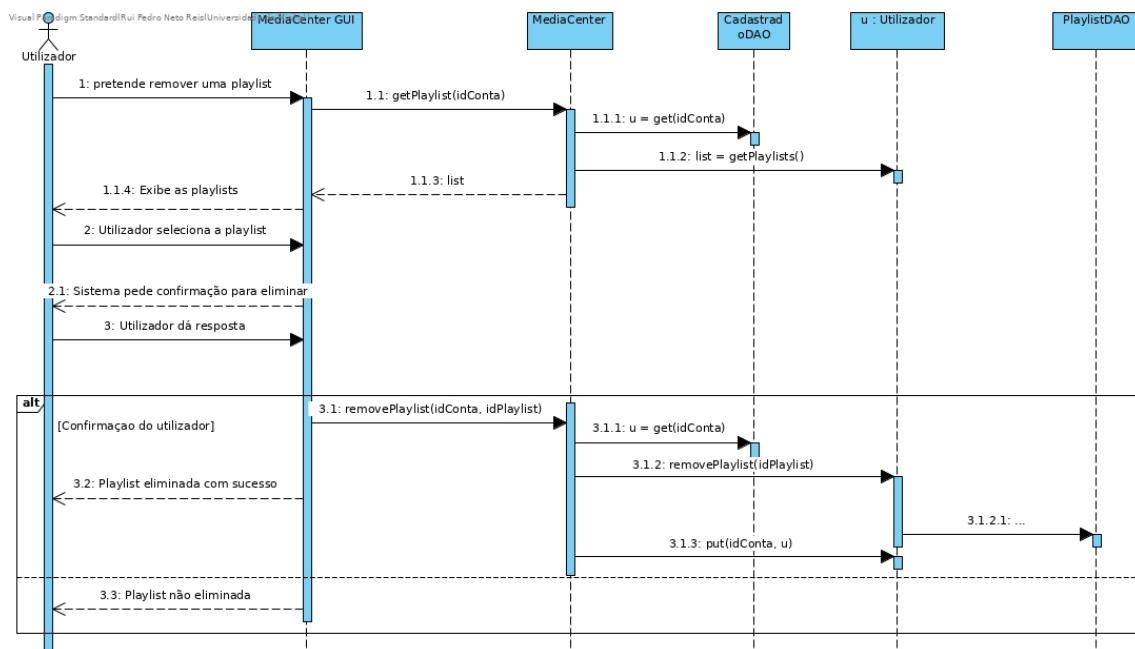


Figura 18: Diagrama de Sequência de Remover Playlist.

### 2.5.11 Editar playlist

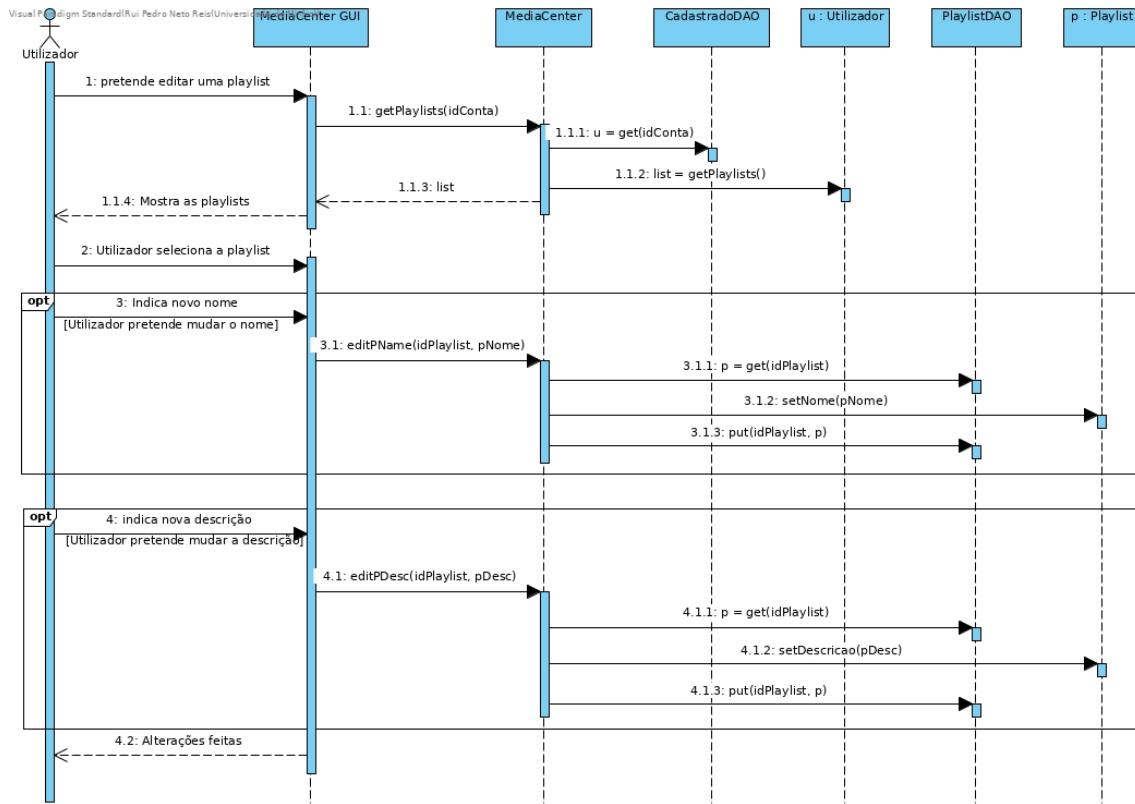


Figura 19: Diagrama de Sequência de Editar Playlist.

### 2.5.12 Reproduzir Conteúdo

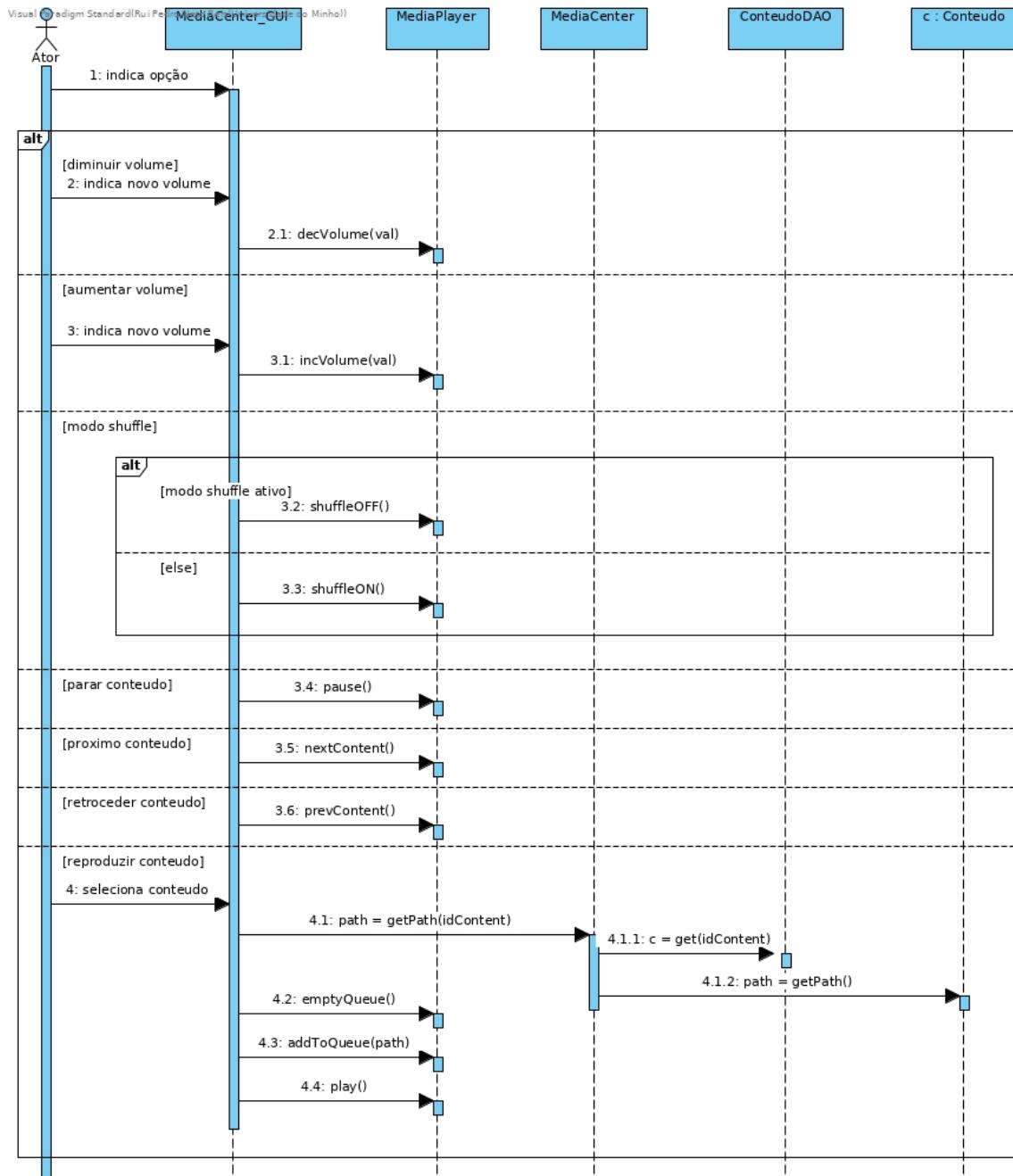


Figura 20: Diagrama de Sequência de Reproduzir Conteúdo.

### 2.5.13 Adicionar Conteúdo à Playlist

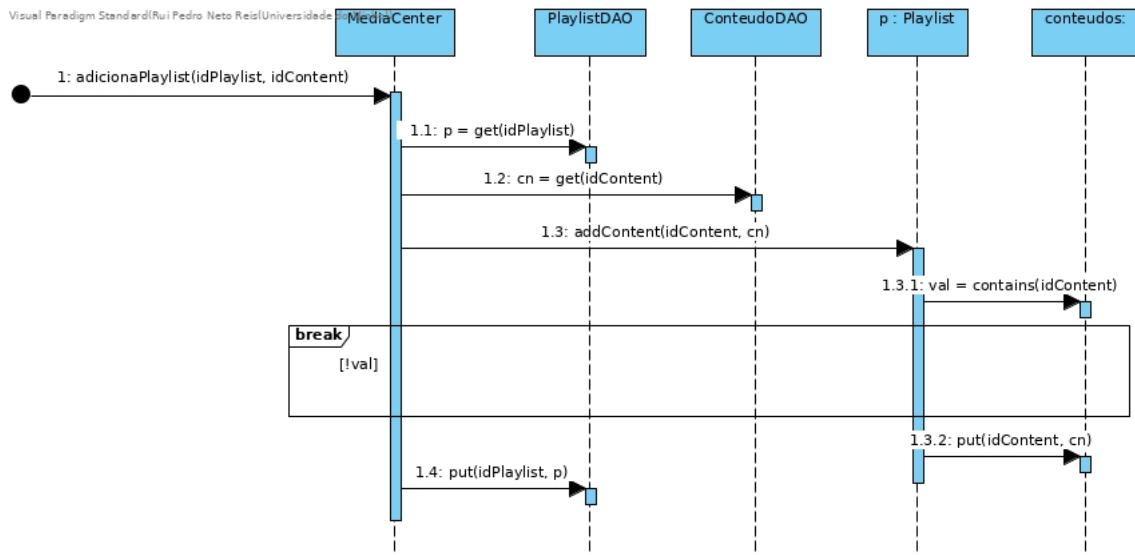


Figura 21: Diagrama de Sequência de Adicionar Conteúdo à Playlist.

### 2.5.14 Remover Conteúdo da Playlist

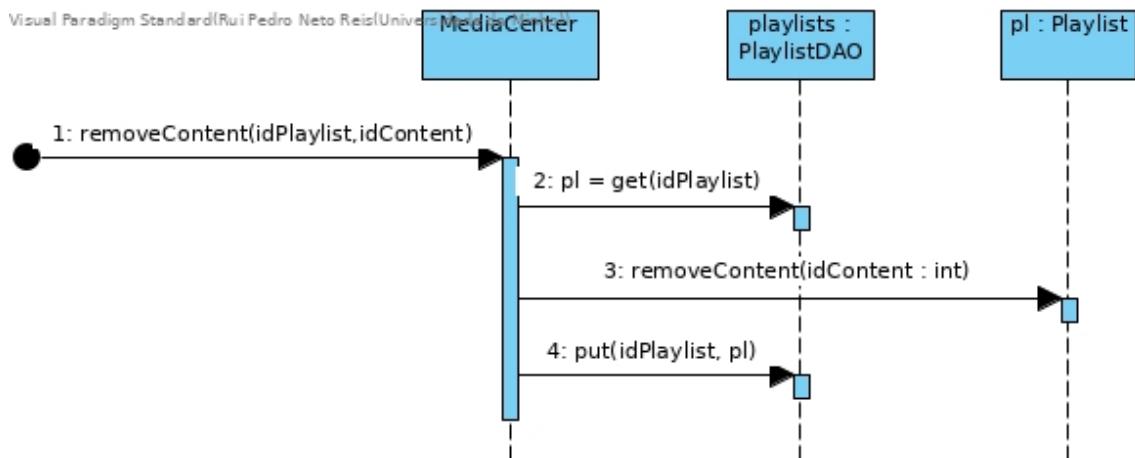


Figura 22: Diagrama de Sequência de Remover Conteúdo da Playlist.

### 2.5.15 Adicionar Álbum à Playlist

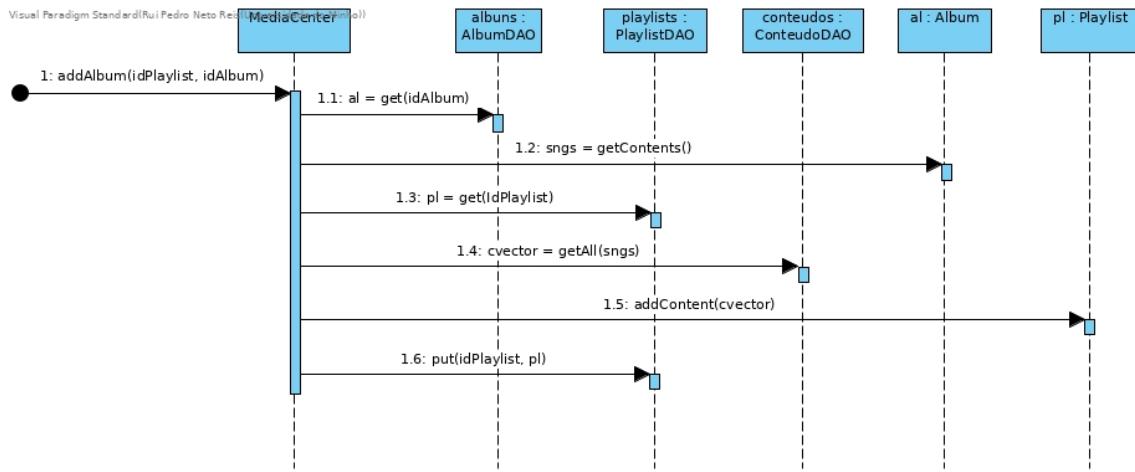


Figura 23: Diagrama de Sequência de Adicionar Álbum à Playlist.

### 2.5.16 Apagar Conteúdo

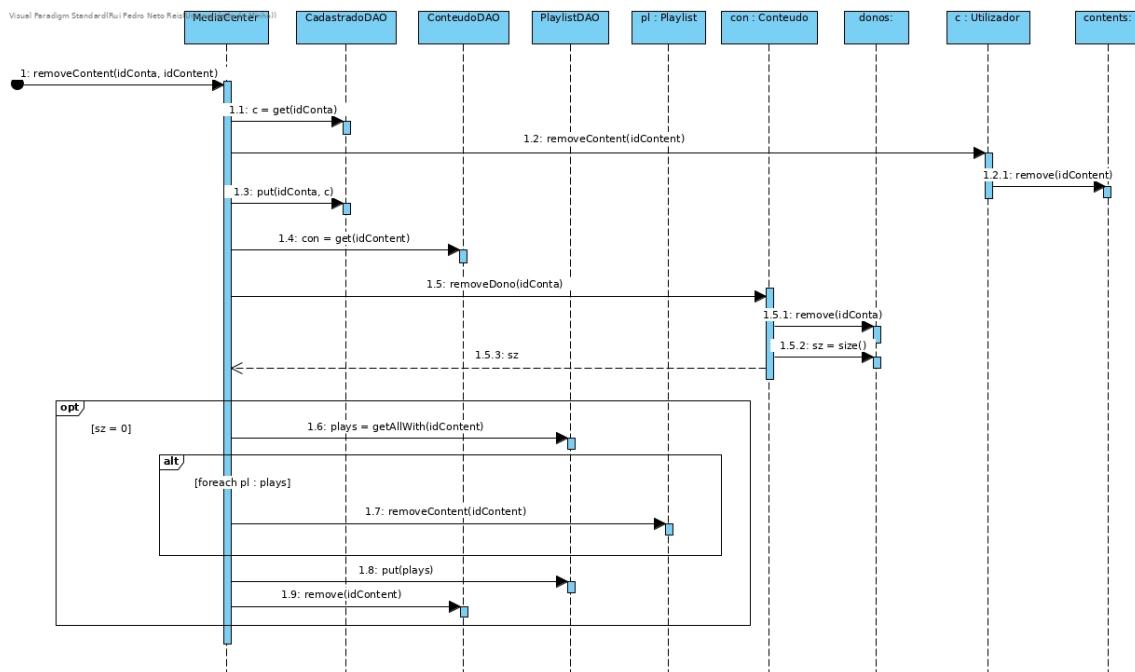


Figura 24: Diagrama de Sequência de Apagar Conteúdo.

### 2.5.17 Editar Utilizador

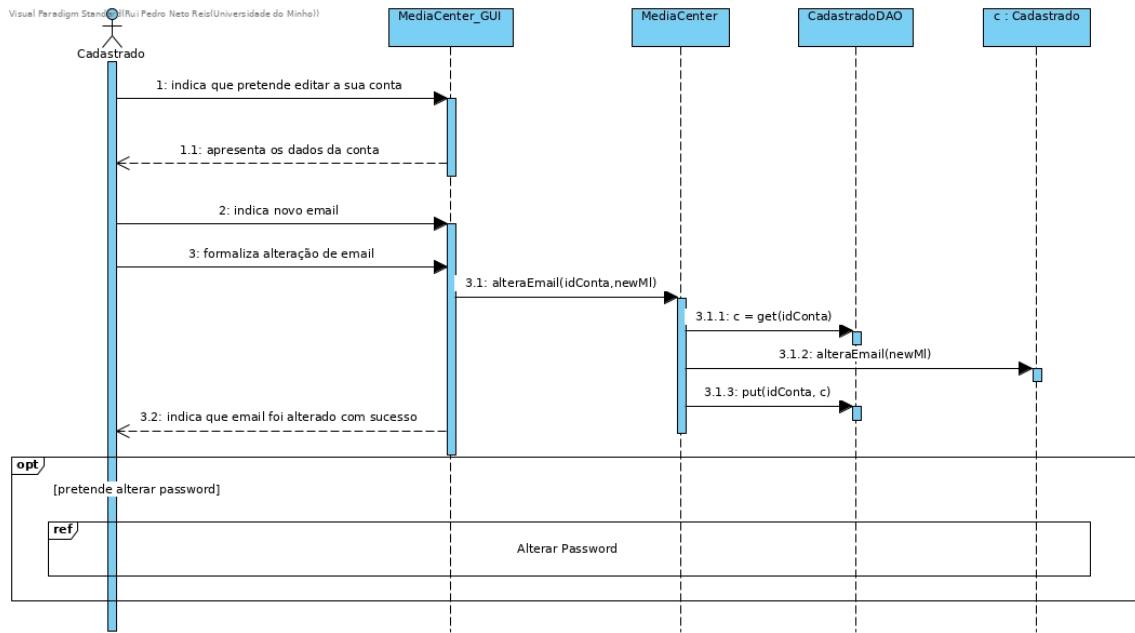


Figura 25: Diagrama de Sequência de Editar Utilizador.

### 2.5.18 Alterar Password

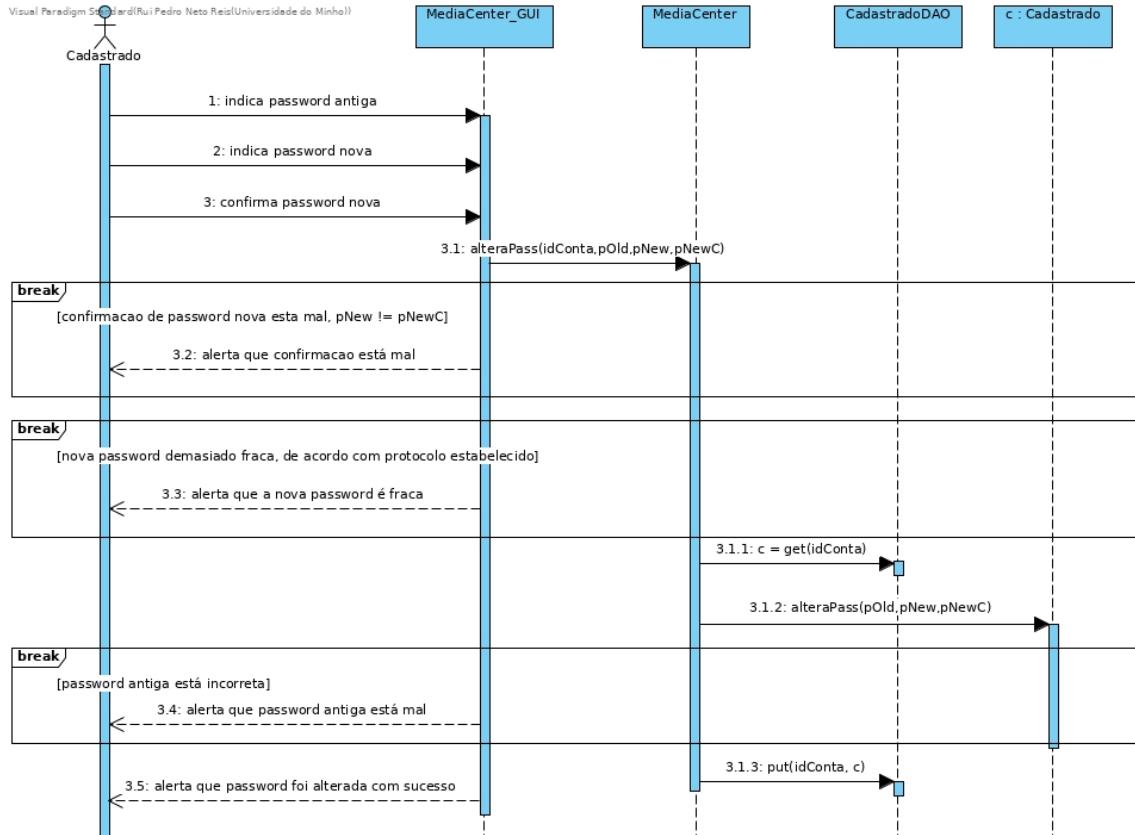


Figura 26: Diagrama de Sequência de Alterar Password.

### 2.5.19 Fazer convite de amigo

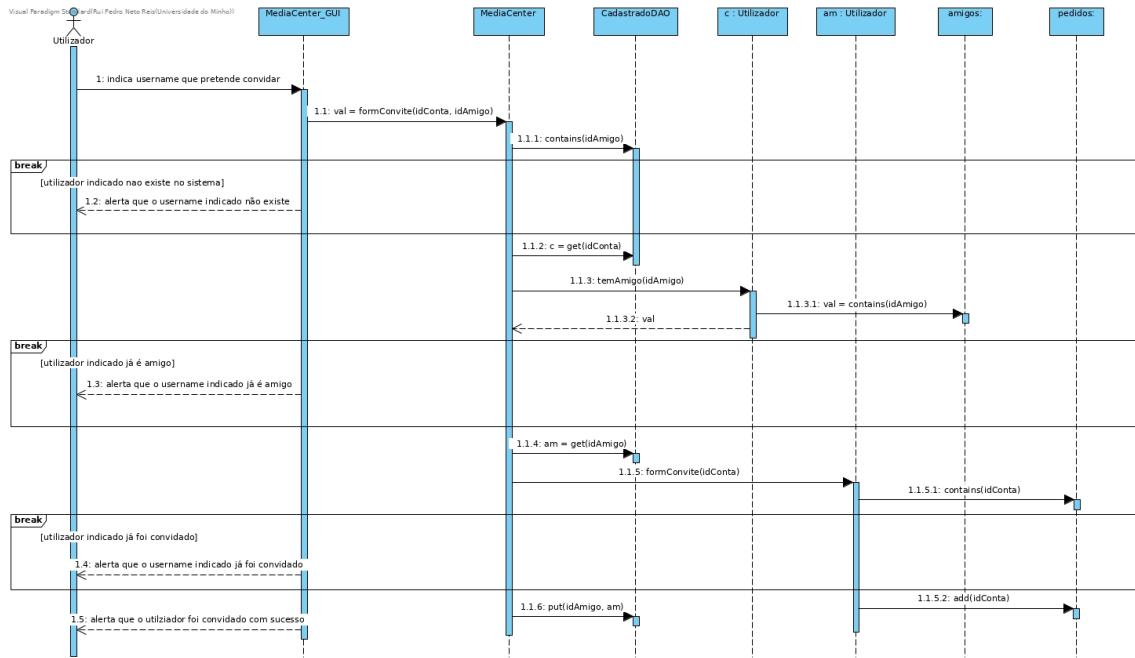


Figura 27: Diagrama de Sequência de Fazer convite de amigo.

### 2.5.20 Responder a convite de amigo

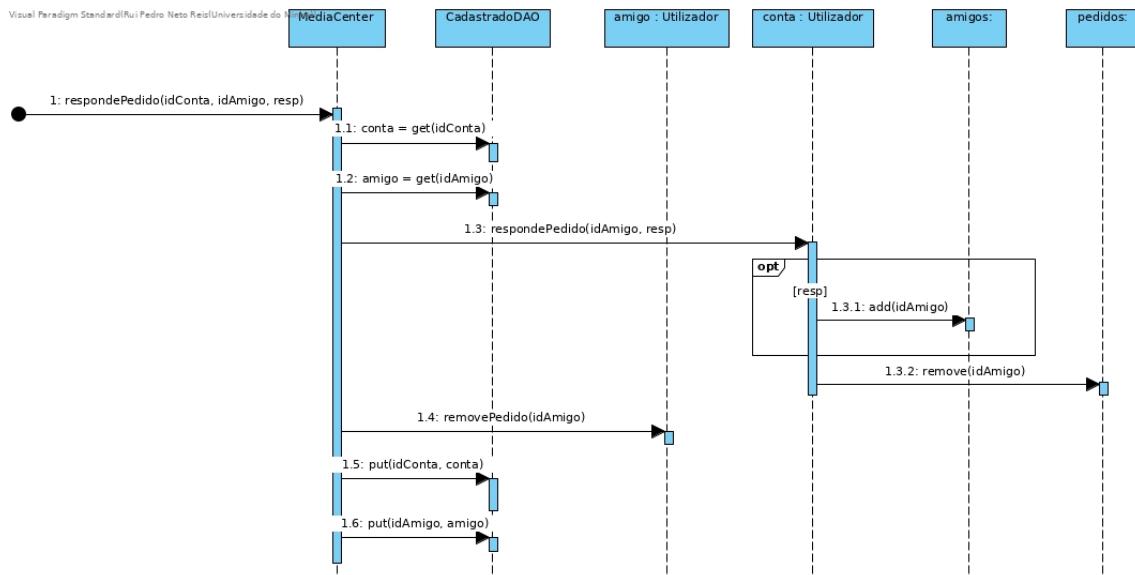


Figura 28: Diagrama de Sequência de Responder a convite de amigo.

### 2.5.21 Remover amigo

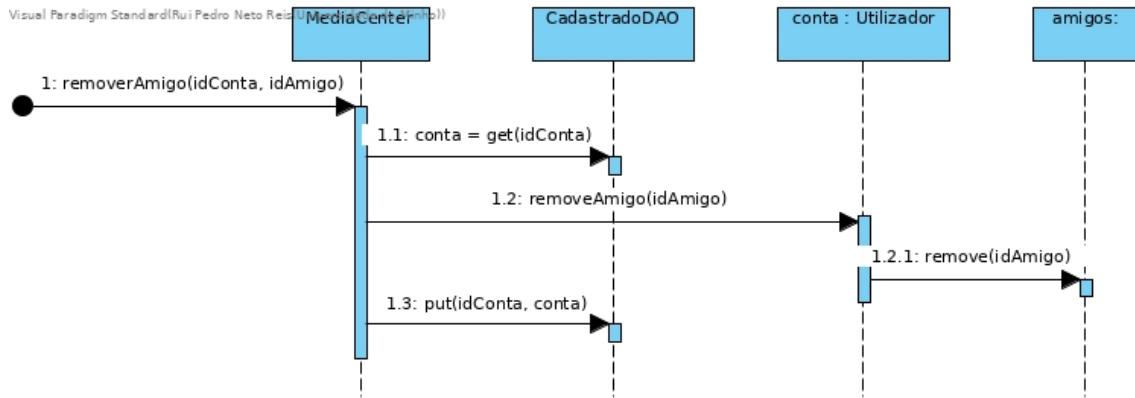


Figura 29: Diagrama de Sequência de Remover amigo.

## 2.6 Diagrama de Packages

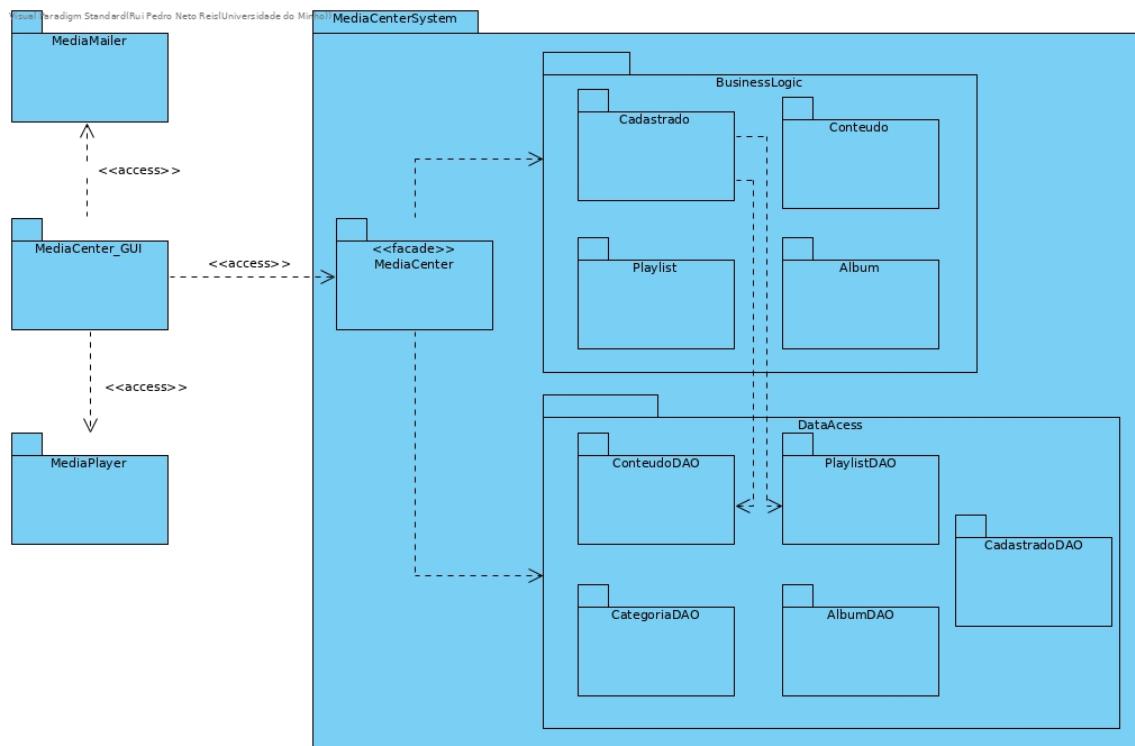


Figura 30: Diagrama de Packages do Sistema.

## 2.7 Protótipo da Interface



Figura 31: Interface de Login.

Media Center				
	Home			
	Nome	Owner	Duração	Actions
	media1	owner1	x:xx	▶ ...
	media2	owner1	x:xx	▶ ...
	media3	owner1	x:xx	▶ ...
	media4	owner3	x:xx	▶ ...
	media5	owner4	x:xx	▶ ...
	media6	owner1	x:xx	▶ ...
	media7	owner1	x:xx	▶ ...
	media8	owner2	x:xx	▶ ...
	media9	owner1	x:xx	▶ ...
	media10	owner4	x:xx	▶ ...
	media11	owner1	x:xx	▶ ...
	media12	owner2	x:xx	▶ ...
	media13	owner1	x:xx	▶ ...
	media14	owner5	x:xx	▶ ...
	media15	owner1	x:xx	▶ ...
	media16	owner1	x:xx	▶ ...
	media17	owner1	x:xx	▶ ...
<b>Categorias</b>				
Categoria 1				
Categoria 2				
Categoria 3				
Categoria 4				
<b>Playlists</b>				
Playlist 1				
Playlist 2				
Playlist 3				
Playlist 4				
Playlist 5				
<b>Amigos</b>				
Amigo 1				
Amigo 2				
Amigo 3				
Amigo 4				
Amigo 5				
<b>+ Menu Amigos</b>				
<b>+ Nova Playlist</b>		◀	▶	✚
		x:xx		x:xx

Figura 32: Interface de Utilizador.

**Media Center**

The screenshot shows a user interface titled "Media Center". On the left, there's a sidebar with a "Logout" button and a "Categorias" section containing five items: "Categoria 1", "Categoria 2", "Categoria 3", "Categoria 4", and "Categoria 5". The main area is a table titled "Home" with columns: Nome, Owner, Duração, and Actions. The data in the table is as follows:

Nome	Owner	Duração	Actions
media1	owner1	x:xx	▶...
media2	owner1	x:xx	▶...
media3	owner1	x:xx	▶...
media4	owner3	x:xx	▶...
media5	owner4	x:xx	▶...
media6	owner1	x:xx	▶...
media7	owner1	x:xx	▶...
media8	owner2	x:xx	▶...
media8	owner1	x:xx	▶...
media9	owner1	x:xx	▶...
media10	owner4	x:xx	▶...
media11	owner1	x:xx	▶...
media12	owner2	x:xx	▶...
media13	owner1	x:xx	▶...
media14	owner5	x:xx	▶...
media15	owner1	x:xx	▶...
media16	owner1	x:xx	▶...
media17	owner1	x:xx	▶...

At the bottom of the main area, there are icons for search, play, and volume, followed by the text "x:xx | x:xx".

Figura 33: Interface de Convidado.

**Alterar Password**

This is a window titled "Alterar Password". It contains two input fields: "Password Antiga:" and "Nova Password:", both represented by masked text boxes. Below these fields is a "alterar" button.

Figura 34: Interface de Alteração de Password.

**Reset Password**

This is a window titled "Reset Password". It contains two input fields: "Username:" and "E-mail:", both represented by text boxes. Below these fields is a "Send" button.

Figura 35: Interface de Reset de Password.

**Upload Media**

Select File ExampleName.mp3 File is Valid/Invalid!

Nome

Artista(s)

Categoria(s)

Album

Upload

This screenshot shows a window titled 'Upload Media'. At the top, there's a 'Select File' button, a file name 'ExampleName.mp3', and a status message 'File is Valid/Invalid!'. Below these are four input fields labeled 'Nome', 'Artista(s)', 'Categoria(s)', and 'Album'. A large 'Upload' button is located at the bottom right.

Figura 36: Interface de Upload de Conteúdo.

**Criar nova playlist**

Nome da Playlist

Descrição

Tipo da Playlist ▼

Aleatório

Categoria

Artista

Simples

Cancelar Confirmar

This screenshot shows a window titled 'Criar nova playlist'. It contains fields for 'Nome da Playlist' and 'Descrição'. A dropdown menu for 'Tipo da Playlist' is open, showing options: 'Aleatório', 'Categoria', 'Artista', and 'Simples'. At the bottom are 'Cancelar' and 'Confirmar' buttons.

Figura 37: Interface de Criação de Nova Playlist.

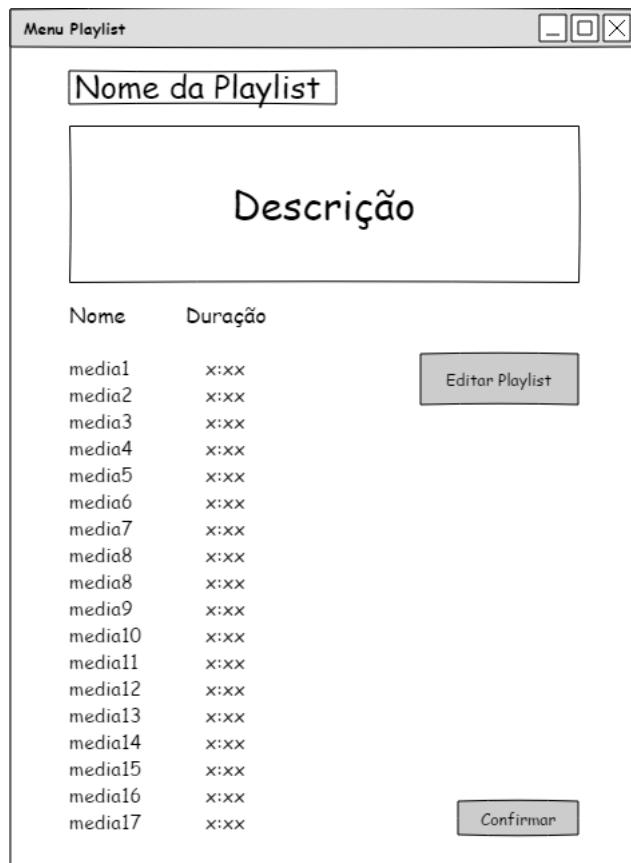


Figura 38: Interface de Gestão da Playlist.

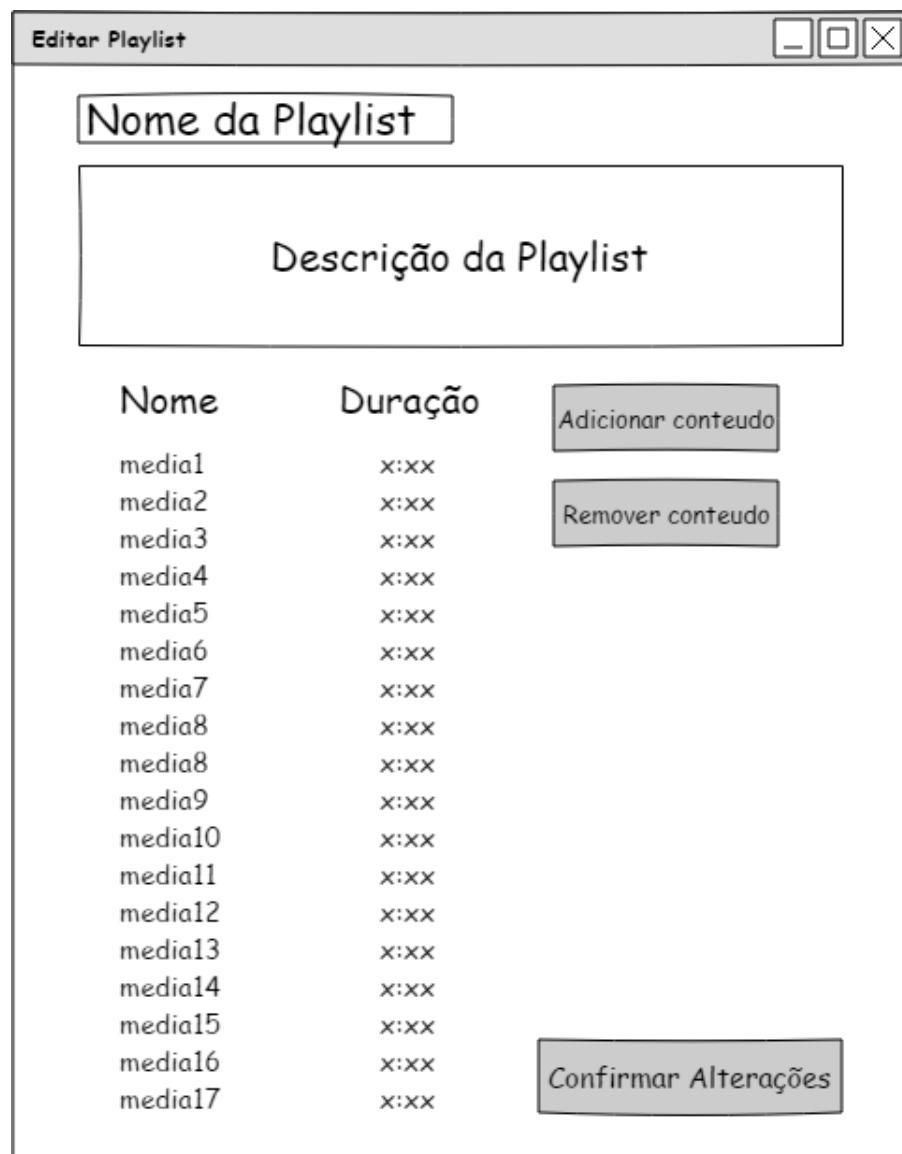


Figura 39: Menu de edição de uma playlist.

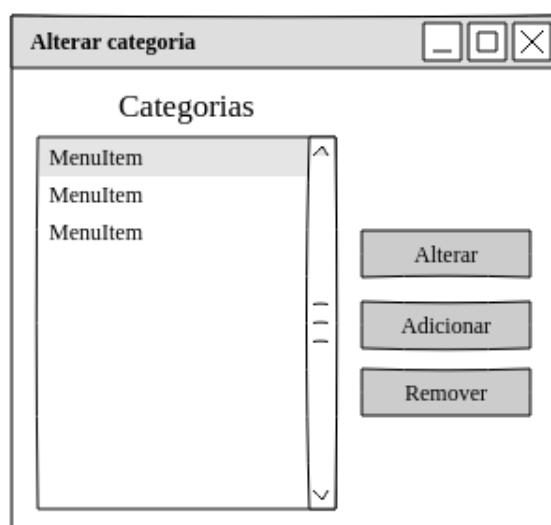


Figura 40: Interface de Gestão de Categorias do Conteúdo.

**Alterar Categoria**

Insira a nova categoria,

**Alterar**

Figura 41: Interface de Alteração de Categoria.

**Admin System**

user1	email1@gmail.com
user2	email2@gmail.com
user3	email3@outlook.com
user4	email4@live.com.pt

**Adicionar**

**Remover**

**Sobre**

Figura 42: Interface de Administração.

**Admin System - Criar Conta**

Username:

Email:

Utilizador       Admin

**Criar conta**

Figura 43: Interface de Criação de Conta.

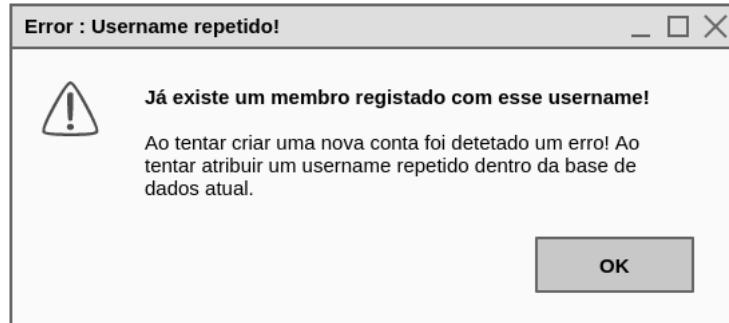


Figura 44: Interface de Erro Básico na Criação de Conta.



Figura 45: Interface de Informação Adicional Sobre uma Conta.

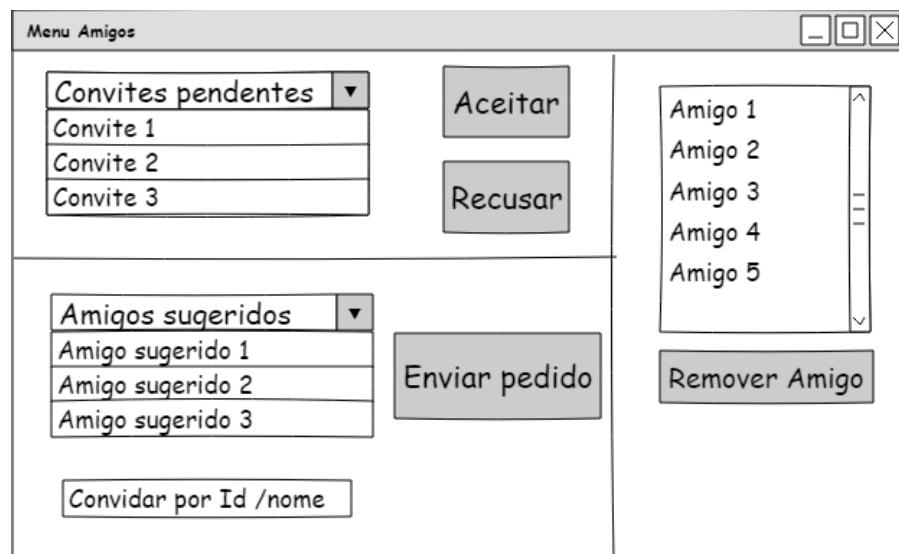


Figura 46: Menu de gestão de amigos.

## 2.8 Diagrama de Estado

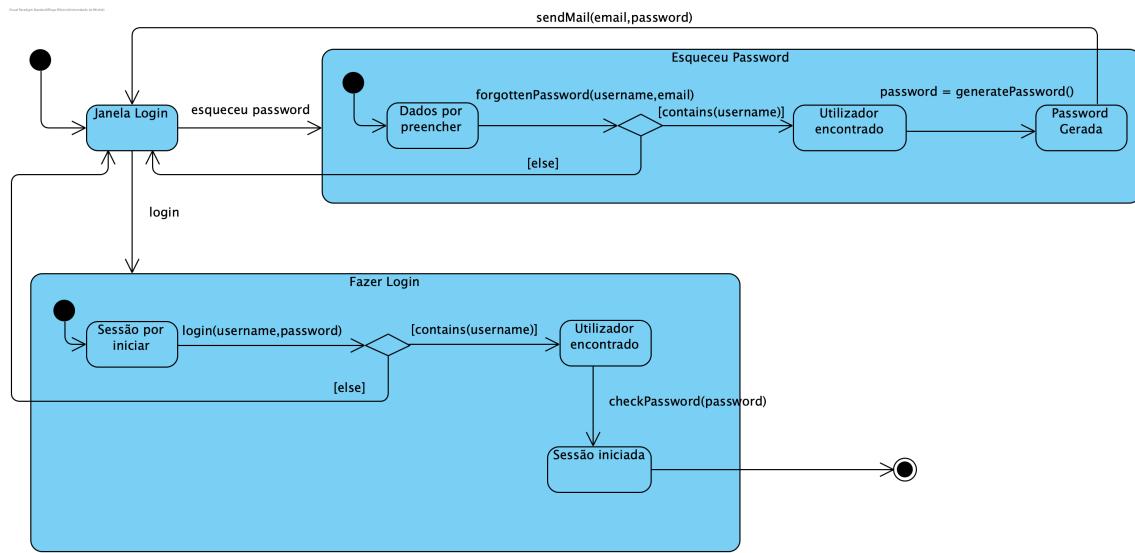


Figura 47: Diagrama de Estado para o login.

## 2.9 Diagrama de Componentes

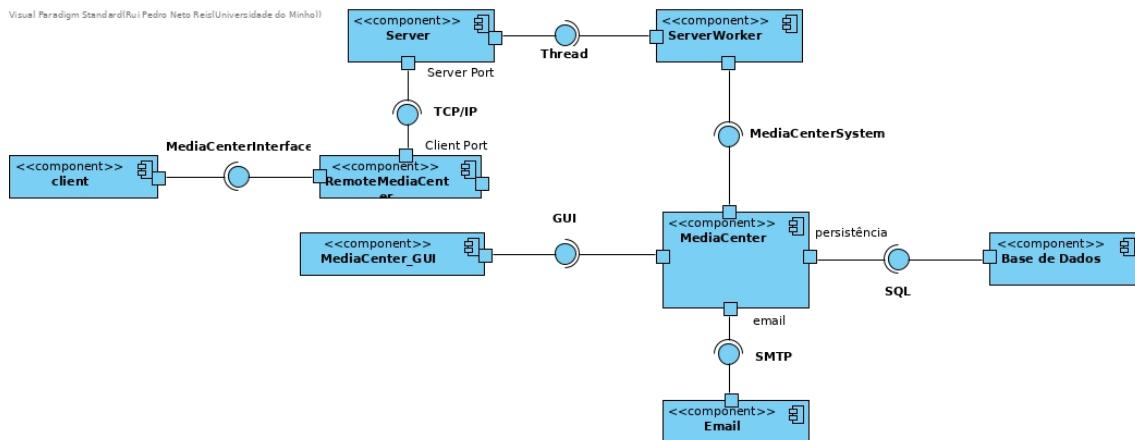


Figura 48: Diagrama de Componentes do Sistema.

## 2.10 Diagrama de Atividade

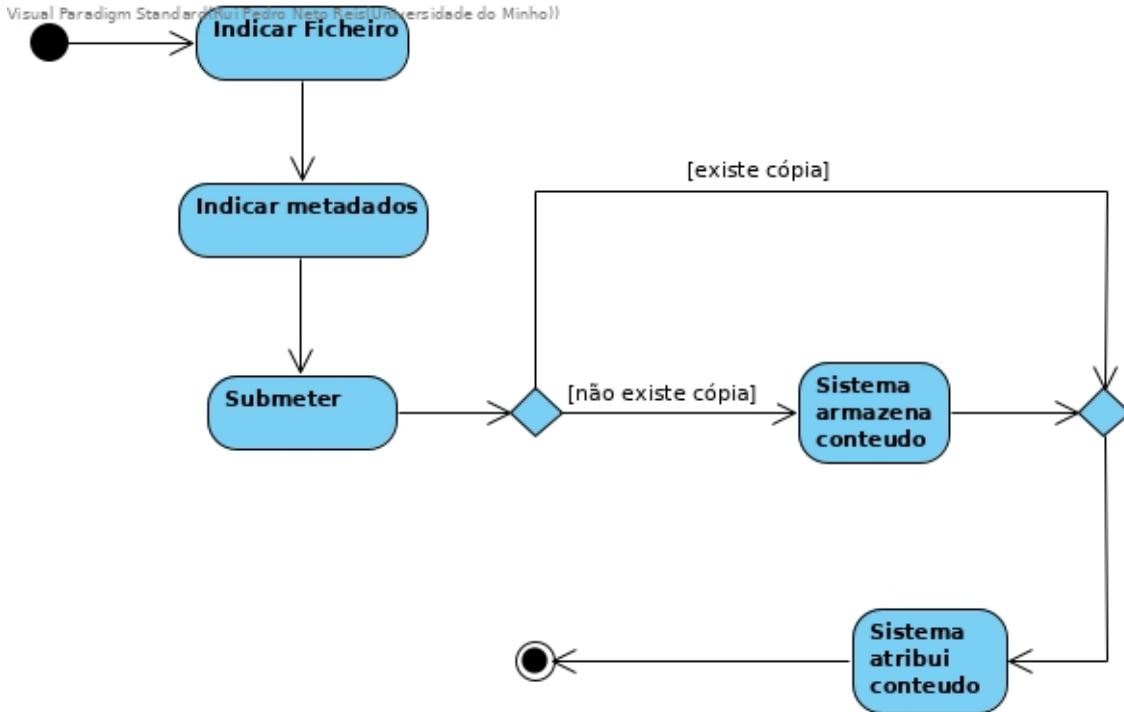


Figura 49: Diagrama de Atividade do Use Case fazer upload.

## 2.11 ORM - Object-Relational Mapping

De modo a garantir a persistência de dados no nosso MediaCenter recorremos ao ORM. Para isso mapeamos as entidades/classes em tabelas e os seus respetivos atributos em colunas das tabelas. Após analisar o nosso Modelo de Domínio e os Diagramas de Sequência identificamos as entidades que deviam persistir, sendo que foram as seguintes: Album, Cadastrado, Categoria, Conteudo e Playlist. Estas entidades foram convertidas em DAOs (Data Access Objects) sendo que também foi necessário mapear o relacionamento entre estas entidades de modo a respeitar o Modelo Relacional.

### 3 Implementação

#### 3.1 JDBC – Java DataBase Connectivity

De modo a isolar a ligação à Base de Dados na Camada de Dados utilizámos uma API para acesso a Bases de Dados para aplicações Java. No nosso caso instalámos o driver para Bases de Dados MySQL (MySQL Connector) pois o MediaCenter deverá utilizar o Modelo Relacional para garantir a persistência de dados.

O driver é inicializado através da seguinte linha de código:

```
Class.forName("com.mysql.cj.jdbc.Driver");
```

A ligação foi implementada da seguinte maneira:

```
private final static String url = "jdbc:mysql://localhost:3306/mydb";
private final static String user = "root";
private final static String pass = "root";

public static Connection makeConnection() throws SQLException {
    return DriverManager.getConnection(url, user, pass);
}
```

Para a implementação de queries optamos por subdividir o processo em várias funções de modo a evitar repetir código desnecessário. Na classe DBAccess estão funções que recebem como parâmetros os valores a colocar nas respetivas queries sql, sendo que cada DAO, como por exemplo o AlbumDAO, chama as respetivas funções das queries pretendidas fornecendo os valores como parâmetro, e recebendo os valores retornados pela query sql. O mesmo se aplica quando um DAO pretende armazenar informação na Base de Dados.

### 3.2 Tabelas do Modelo Relacional

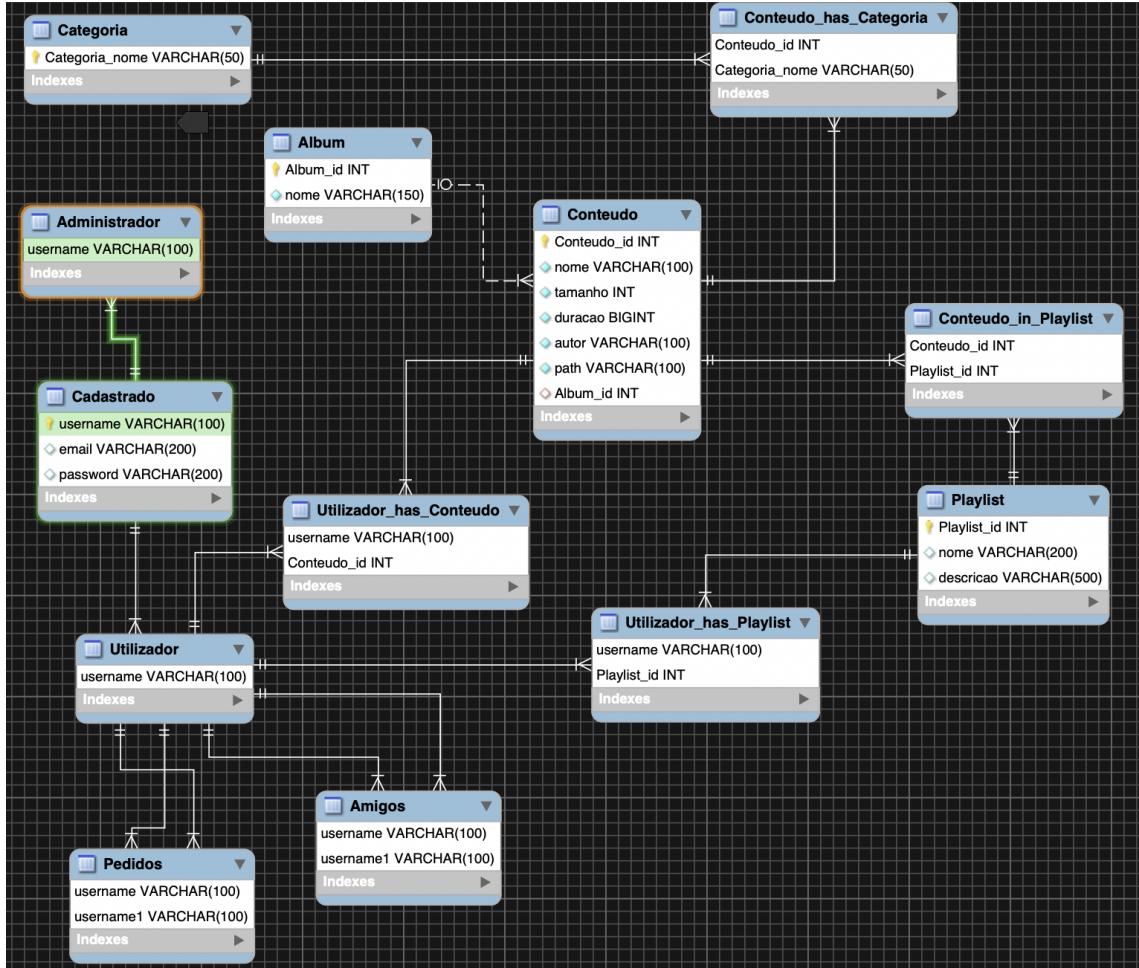


Figura 50: Diagrama ER

### 3.3 Detalhes Relevantes da Implementação

#### 3.3.1 Implementação Inicial

Na fase de modelação deste Media Center foram especificados diversos Use Cases que representam aquilo que um utilizador espera que o sistema seja capaz de fazer. Após toda a fase de modelação e após garantirmos a persistência dos dados no nosso sistema seguiu-se a fase de implementação destes mesmos UCs. Todos os UCs foram implementados com sucesso, sendo que isto se traduz num Media Center que cumpre todos os requisitos previamente definidos.

Apesar da implementação com sucesso destes UCs, convém notar que estes se encontram numa fase inicial, isto é, a implementação deste Media Center é uma implementação inicial. Esta implementação ainda não se encontra completamente otimizada, sendo que existem aspetos a melhorar tanto na sua eficiência, bem como na sua segurança.

#### 3.3.2 Utilização do VLC

Para a reprodução de media foi-nos sugerido a utilização do VLC. O VLC possui uma interface gráfica e também possui uma CLI (Command Line Interface), sendo que optámos por utilizar a segunda opção. A nossa escolha deveu-se ao facto de assim podermos esconder a nossa implementação do VLC de modo a que o utilizador convencional não saiba o que está a acontecer. A nossa implementação do VLC ficou contida numa classe a que chamamos de MediaPlayer. Na classe MediaPlayer, o contrutor por defeito simplesmente inicializa um processo em background do VLC na sua versão CLI. De seguida, toda a comunicação com o VLC é feita a partir de um

InputStream e de um OutputStream. Todas as funções desta classe escrevem para o InputStream deste processo o comando que é pretendido. sendo que neste modo o utilizador não se depara com nenhum detalhe da implementação.

### 3.3.3 Funcionalidade em Rede

A possibilidade de o Media Center funcionar em rede foi-nos proposta como um desafio extra à implementação, sendo que optamos por seguir esse desafio, e, implementar essa funcionalidade. Através dos conhecimentos adquiridos à UC de Sistemas Distribuídos fomos capazes de implementar um sistema Cliente/Servidor para o nosso Media Center. O nosso sistema é constituído por um Servidor que tem a capacidade de atender múltiplos Clientes. A comunicação é feita através dos Sockets do Java e toda a comunicação, à excepção da transferência dos ficheiros é feita com Strings.

### 3.3.4 Reprodução de Conteúdo

O ponto anterior levantou uma questão que causou algum debate. Esta questão centra-se na reprodução de um conteúdo. Após algum debate chegamos a um consenso e decidimos que para reproduzir conteúdo era preciso transferir o ficheiro localmente.

Para conseguir isso seguimos a seguinte metodologia: sempre que um utilizador adiciona um conteúdo à fila de reprodução(queue), esse ficheiro é descarregado para uma pasta na diretoria do Cliente denominada de .reproduction. Depois de descarregado o conteúdo é possível a sua reprodução. Esta pasta é uma pasta invisível ao utilizador com a utilização do . antes do nome, no entanto esta implementação pode muito facilmente ser contornada por um utilizador mais experiente. Este é um dos pontos da implementação que no futuro seria alterado e melhorado. Uma das ideias que também surgiu quando debatemos a ideia foi a gestão do armazenamento, isto é, quando um utilizador fecha o Media Center, todo o conteúdo da pasta .reproduction é eliminado. Esta funcionalidade encontra-se presente no MediaCenter.

A conclusão a que chegamos é que esta implementação de reprodução de conteúdo necessita de ser revista e alterada no futuro de modo a ser mais eficiente e segura.

## 3.4 Descrição da Interface

A Interface que por nós produzida foi obtida usando o GridLayout. Também recorremos a uma ferramenta mais gráfica para gerar certos elementos da interface e compreender melhor como funciona o GridLayout. Para esse efeito recorremos a um programa chamado Jviper, que é um designer gráfico de interfaces Java e Swing. Comparando os resultados obtidos com as nossas propostas para a interface final, podemos dizer que o resultado foi o esperado, tanto a nível de funcionalidade, bem como a nível de layout, sendo que à exceção de uma correção ou outra, fomos capazes de replicar os protótipos.

No formulário de login existem 2 campos de input para o username e para a password. Para além disso existem 3 botões: o botão de Login para efetuar login como cadastrado, o botão Login as Guest para efetuar login como convidado, e, o botão Forgot Password para o caso de um cadastrado querer repor a sua password.

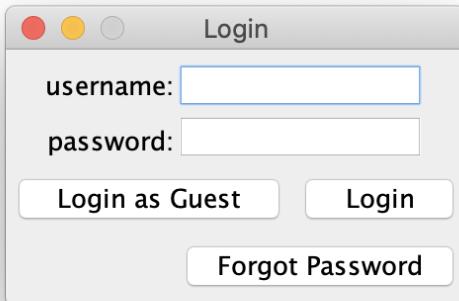


Figura 51: Formulário de Login.

Na vista principal de um cadastrado existem 3 grandes conjuntos. Do lado esquerdo do separador existe uma aba que contém o username, um botão para efetuar logout, contém as listas de categorias, playlists e amigos da conta em questão, um botão para aceder ao menu de amigos, e, um botão que abre o menu para criar uma playlist nova. Do lado direito possuímos a tabela que exibe o conteúdo em questão (por categoria ou playlist), e no fundo da tabela possuímos a barra de reprodução. A barra de reprodução contém os botões de play/pause, o botão do modo shuffle, e o botão para adicionar conteúdo.

Convém notar que através do uso de ActionListeners e MouseListeners do Java foi-nos possível acrescentar funcionalidades, nomeadamente a possibilidade de abrir o menu de edição de playlist ao clicar duas vezes num item da lista de playlists, ou abrir um JPopupMenu quando se clica no botão de opções de um determinado conteúdo.

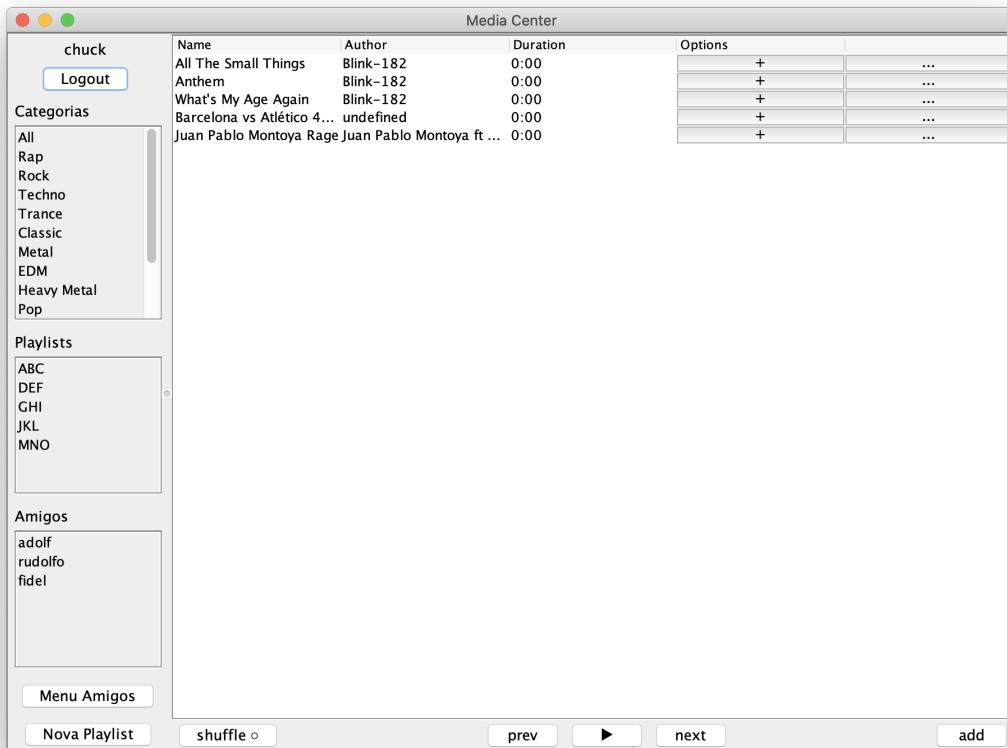


Figura 52: Vista principal de um cadastrado.

O formulário de upload é constituído por 5 campos de input. O primeiro é um JFileChooser que apenas nos permite selecionar ficheiros de um determinado tipo (aqueles que o filtro aplicado permitir). Os restantes são campos para o nome do conteúdo, o(s) artista(s), categoria(s) e álbum. No final existe um botão Upload que submete o formulário.

The screenshot shows a modal dialog box titled 'Upload'. It contains a 'Select File' button at the top, followed by four input fields labeled 'Nome', 'Artista(s)', 'Categoría(s)', and 'Álbum'. At the bottom right is a large 'Upload' button.

Figura 53: Formulário de Upload.

O menu para alterar a categoria de um conteúdo possui uma JList que contém as categorias de um determinado conteúdo, e possui 3 botões: um botão Alterar, que nos permite alterar uma

categoria previamente selecionada na lista, o botão Adicionar, que nos permite adicionar uma categoria ao conteúdo, e, o botão Remover, que nos permite remover uma categoria previamente selecionada.

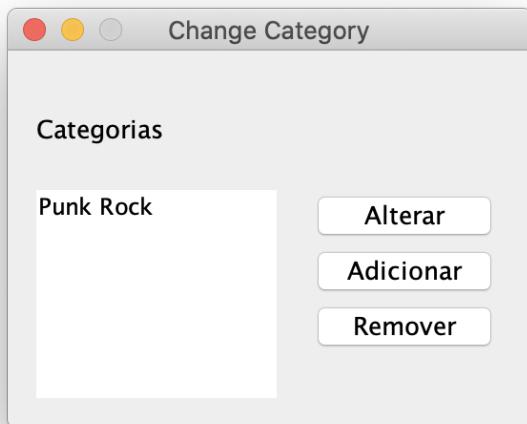


Figura 54: Alterar Categoria de um Conteúdo.

## 4 Conclusões

O trabalho permitiu-nos desenvolver competências na elaboração de um projeto, planeando antecipadamente várias interações que se pretende realizar na aplicação, no âmbito de reduzir a reformulação de código, especialmente, com o desenvolvimento dos diagramas de sequência e dia-grama de classes.

Igualmente, desenvolve-mos competências no que toca à parte gráfica do trabalho, visto que tive-mos que implementar interfaces, que tornam a experiência da aplicação mais "user friendly".

Durante o trabalho surgiram algumas dificuldades, principalmente, na fase da modelação dos dia-gramas de sequência e de componentes. Igualmente, ao especificar os use cases houve uma pequena dificuldade em tornar estes significativos, mas facilmente foi ultrapassada. Na terceira fase do pro-jeto, vimos dificuldade na alteração do diagrama de classes para implementar os DAOs, tendo que refazer o diagrama de classes para permitir uma maior eficiência. Seguidamente, ao implementar OCL surgiu dúvidas sobre a linguagem em si.

Finalmente, cumprindo os requisitos mínimos do trabalho, bem como, a implementação de mais funcionalidades, tais como ser capaz de funcionar em rede, sentimos que está-mos contentes com o projeto.