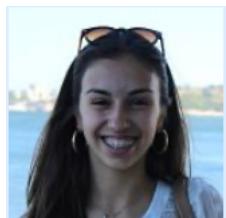


Universidade do Minho

Departamento de Informática

Desenvolvimento de Sistemas Software  
Media Center - Fase 3  
Grupo Nº 25



Mafalda Costa (A83919)



Pedro Fernandes (A84313)



Gonçalo Almeida (A84610)



Lázaro Pinheiro (A86788)

# Introdução

O presente trabalho prático tem como objetivo desenvolver um *media center* para partilha de música/vídeo (*media*) num apartamento de alunos de MIEI onde podem fazer uso da *media* disponível.

Os cenários apresentados para desenvolver o projeto pretendido foram tidos como referência e como requisitos essenciais para a arquitetura da aplicação. Como estes foram idealizados pela equipa docente da unidade curricular, necessitou-se de esclarecer alguns constructos e possíveis itinerários de implementação para alcançar os objetivos.

Este trabalho foi planeado em 3 fases, sendo uma mais valia na consolidação de aprendizagem, aperfeiçoamento e desenvolvimento da aplicação ideal, tratando-se de uma *on-going*.

Esperamos com este trabalho concluir com sucesso o *media center*.

# Modelação

Os modelos são simplificações da realidade, isto é, representações abstratas de um sistema efetuadas através de um determinado ponto de vista. Estes revelam-se úteis para descrever e analisar os problemas e as soluções a desenvolver.

A modelação reveste-se de vantagens na utilização dos seus diversos modelos, na medida em que auxiliam a compreensão da realidade, ajudam a comunicar as ideias de forma simplificada, ajudam a documentar as decisões durante o desenvolvimento.

Assim, é possível descrever o que é considerado essencial num dado contexto, permitindo comunicar apenas os aspetos pretendidos.

## Modelo de domínio

O modelo de domínio captura as entidades (“substantivos” na descrição) dos problemas e os relacionamentos entre eles. Além disso, captura o vocabulário do domínio do problema (glossário de termos), fornece uma *Framework* conceptual (ajudado a pensar) e uma visão estratégica do problema.

Este modelo é estático, pois não representa fluxos de dados, apenas representa o problema não incluindo o *software* a desenvolver.

O modelo de domínio alcançado na 1<sup>a</sup> fase do projeto, foi considerado em parte pois nas fases sucessivas foi pedido para descorar alguns relacionamentos e entidades encontradas, pelo que o modelo apresentado na figura 1 é o modelo de domínio da arquitetura atual.

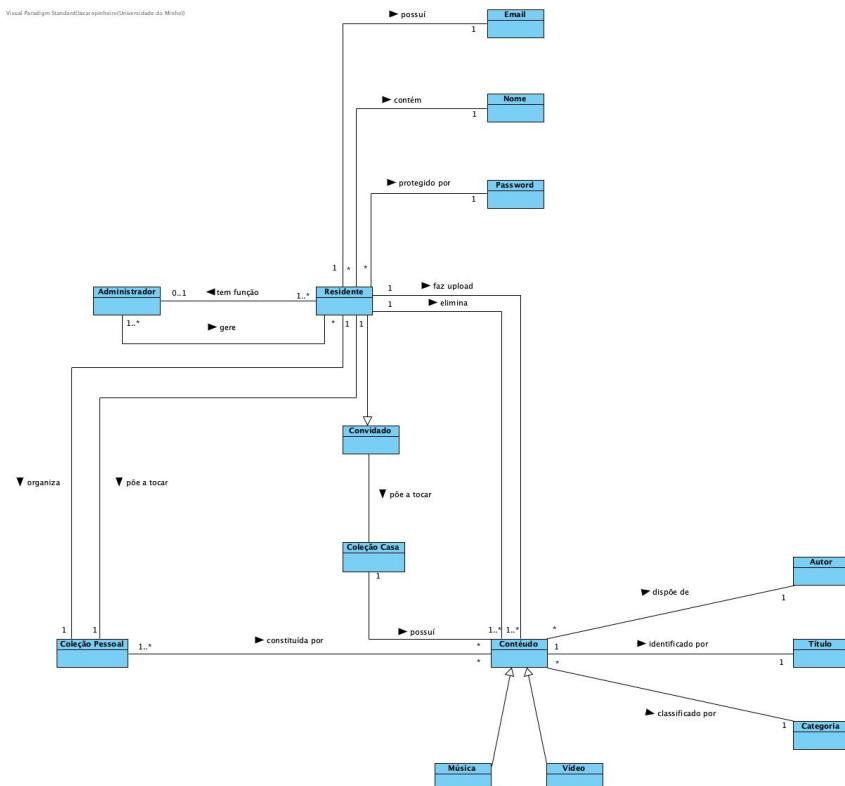


Figura 1 - Modelo de Domínio do Media Center.

## Modelo de Use Cases

O modelo de Use Cases descreve as sequências de interações entre atores externos e um sistema em projeto, para que um dado cenário de utilização seja suportado. Cada passo de interação ator-sistema descrito num Use Case designa-se por evento, ação ou operação, e devem distinguir-se quanto à sua origem (ator ou sistema).

Um Use Case descreve um fluxo principal de eventos/operações (fluxo principal), bem como outros possíveis fluxos ou caminhos (fluxos alternativos) e fluxos que conduzem a situações de insucesso (fluxos de exceção). Os Use Cases devem ser simples e legíveis, não deve conter detalhes sobre a interface com o utilizador e devem ter o nível de detalhe necessário a cada iteração de requisitos.

No decurso do trabalho prático fomos obtendo os Use Cases necessários para modelar a interação do ator com o *media center* (ver Anexo 1).

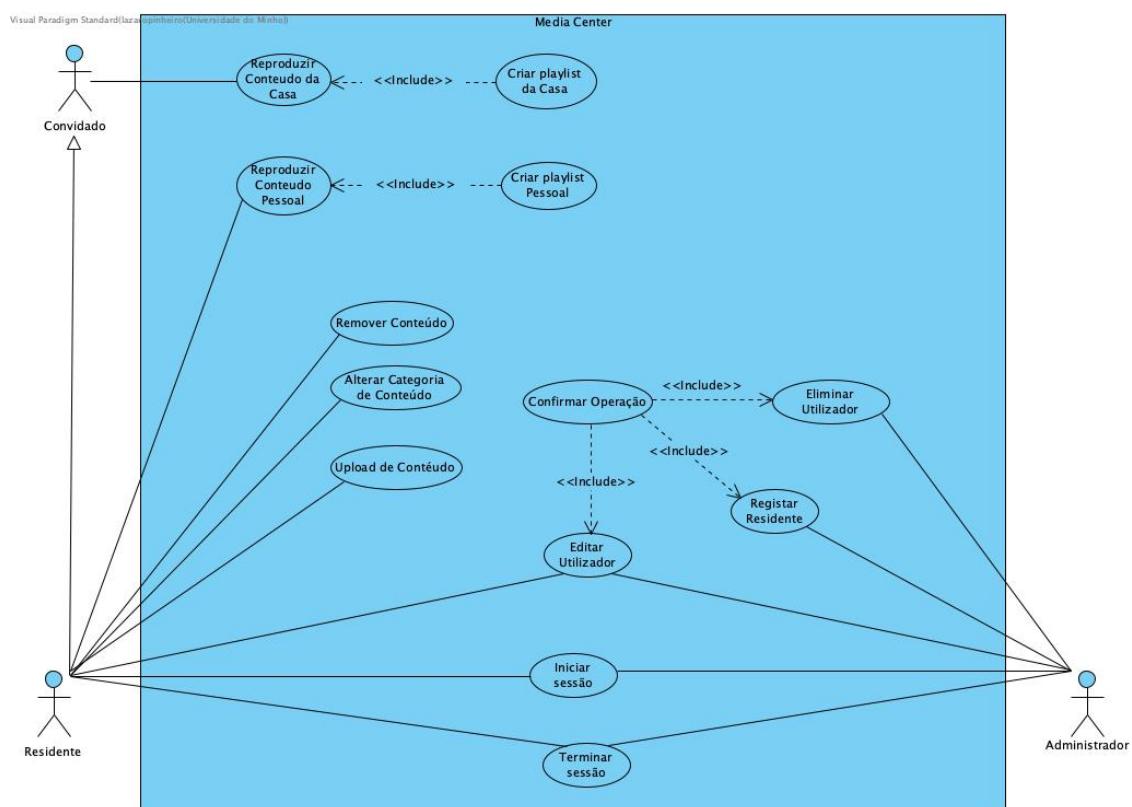


Figura 2 - Diagrama de Use Cases do media center.

## Protótipo da Interface

O protótipo é utilizado para testar um conceito, comunicar e explorar as alternativas e tecnologias.

Neste trabalho, o mesmo foi desenhado numa fase primordial (antes de codificar), com o objetivo de tomar decisões ao nível gráfico e facilitar a interação do ator com o sistema.

Media Center

Email

Password

\*\*\*\*\*

Login

Figura 7 - Protótipo de UI de Login

Media Center

Email

Nome

Password

Tipo de Conta

Cancelar

Criar

Media Center

Selecionar Coleção

Selecionar o Conteúdo

Random

Música 1  
Música 2  
Vídeo 1  
Música 3  
Vídeo 2  
...

Figura 4 - Protótipo de UI de Reproduzir Conteúdo.

Figura 5 - Protótipo de UI de Registrar Utilizador.

Media Center

Selecionar o Conteúdo

Música 1  
Música 2  
Música 3  
Música 4  
...

Selecionar a nova categoria

text goes here

Media Center

Selecionar Ficheiro

Nome do Ficheiro

Upload

Figura 3 - Protótipo de UI de Upload de Conteúdo.

Figura 6 - Protótipo de UI de Alterar Categoria de Conteúdo.

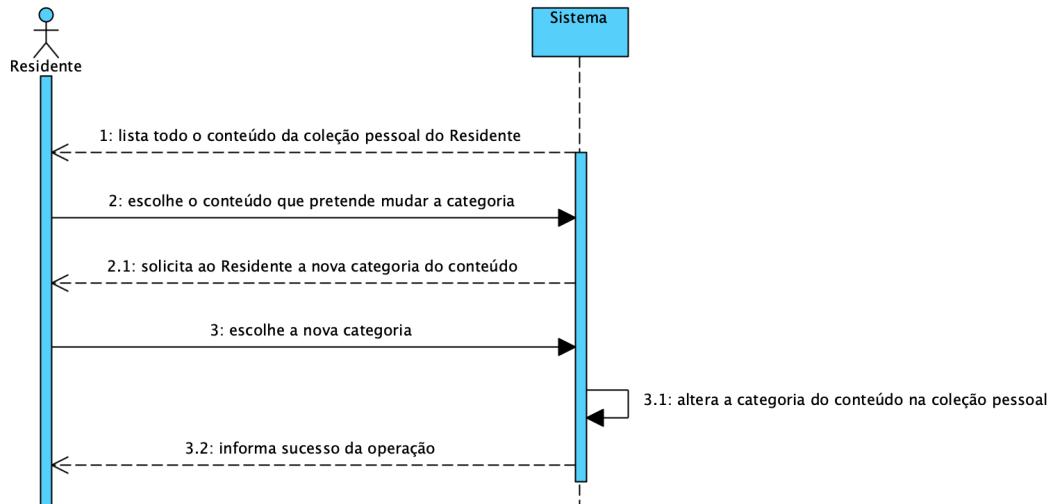
## Use Cases Fase 2

Considerando que nesta fase era desconhecido o armazenamento da informação numa base de dados relacional, os diagramas obtidos nesta 2<sup>a</sup> fase corroboram os da 1<sup>a</sup> fase (cf. anexo 1) com acréscimo de novos use cases.

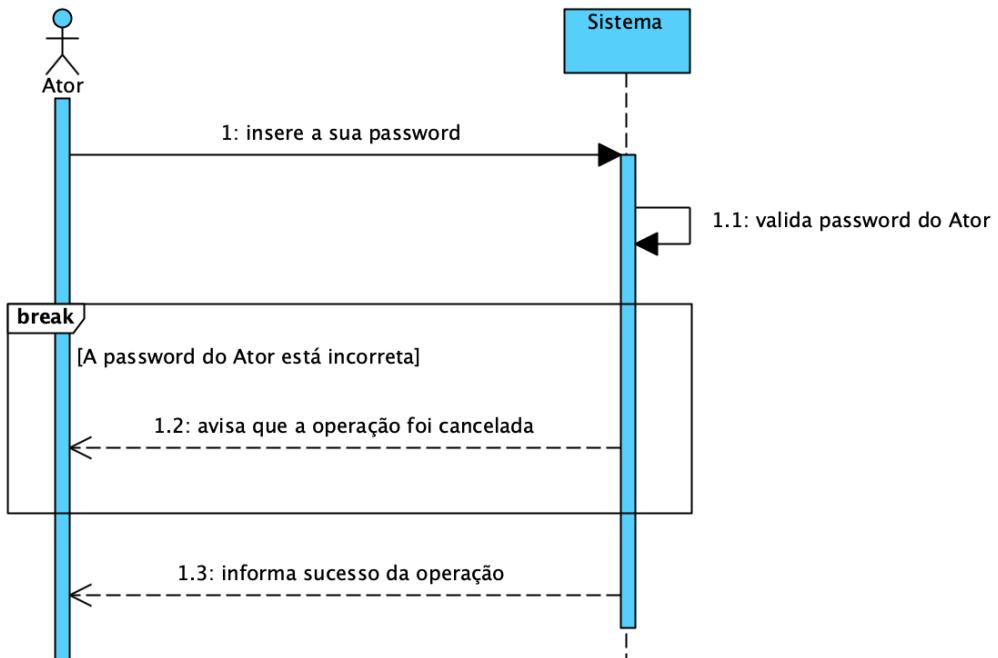
Assim, foram identificados os diagramas apresentados de seguida.

*Diagramas de Sequência com subsistemas para cada Use Case*

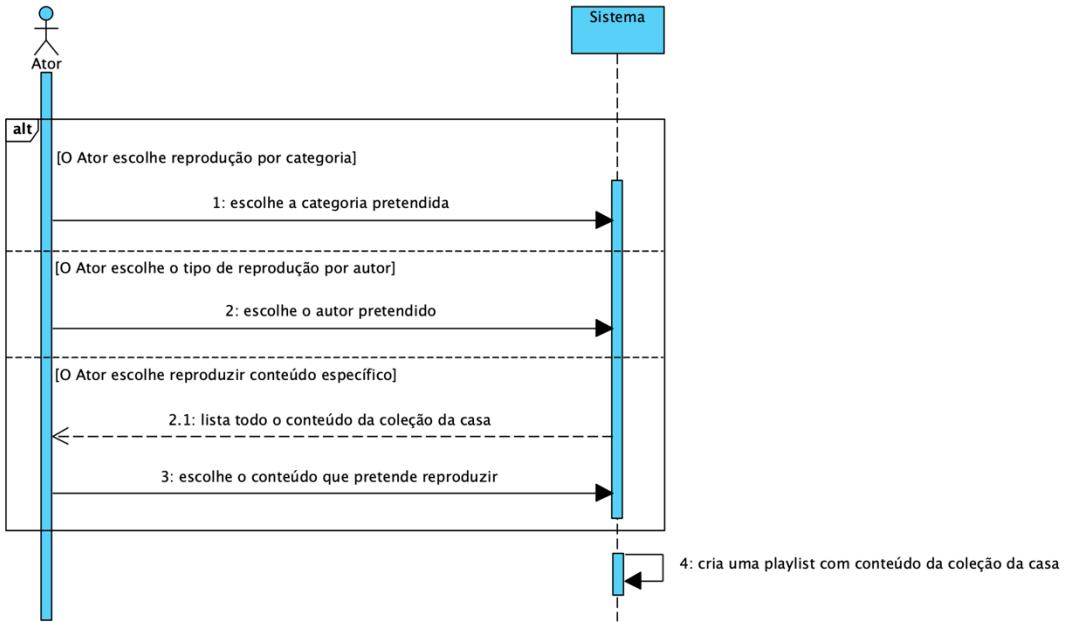
### sd Alterar Categoria de Conteúdo Sequence Diagram



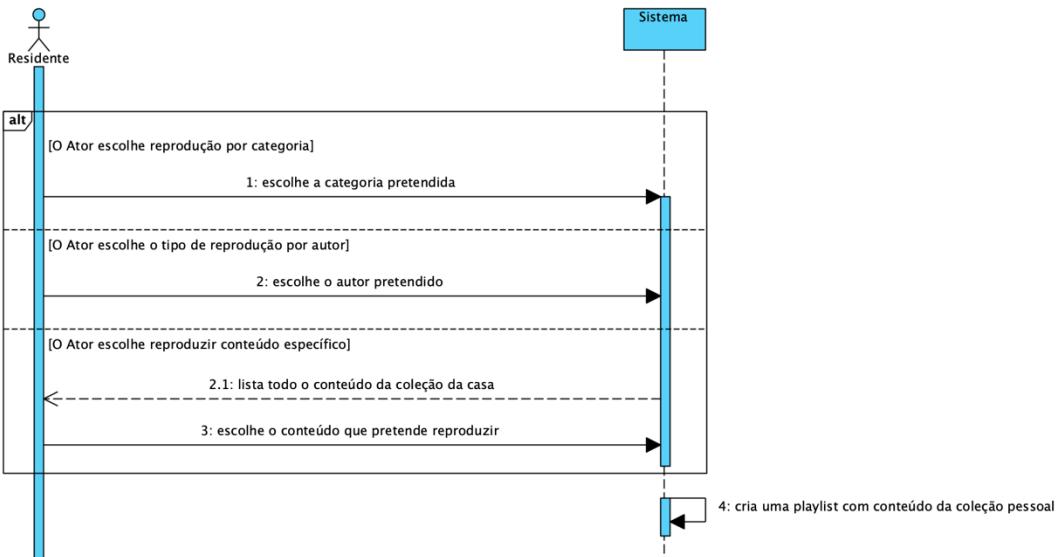
### sd Confirmar Operação Sequence Diagram



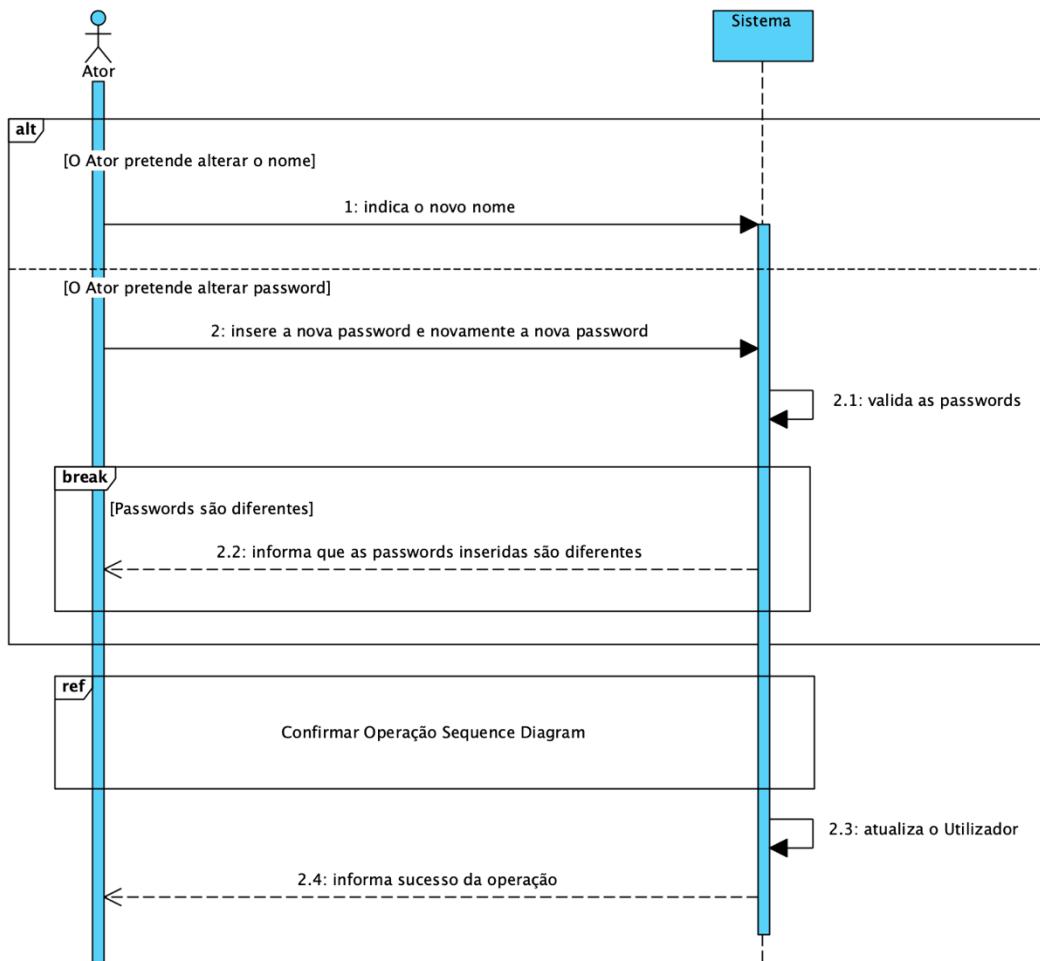
**sd Criar Playlist da Casa Sequence Diagram**



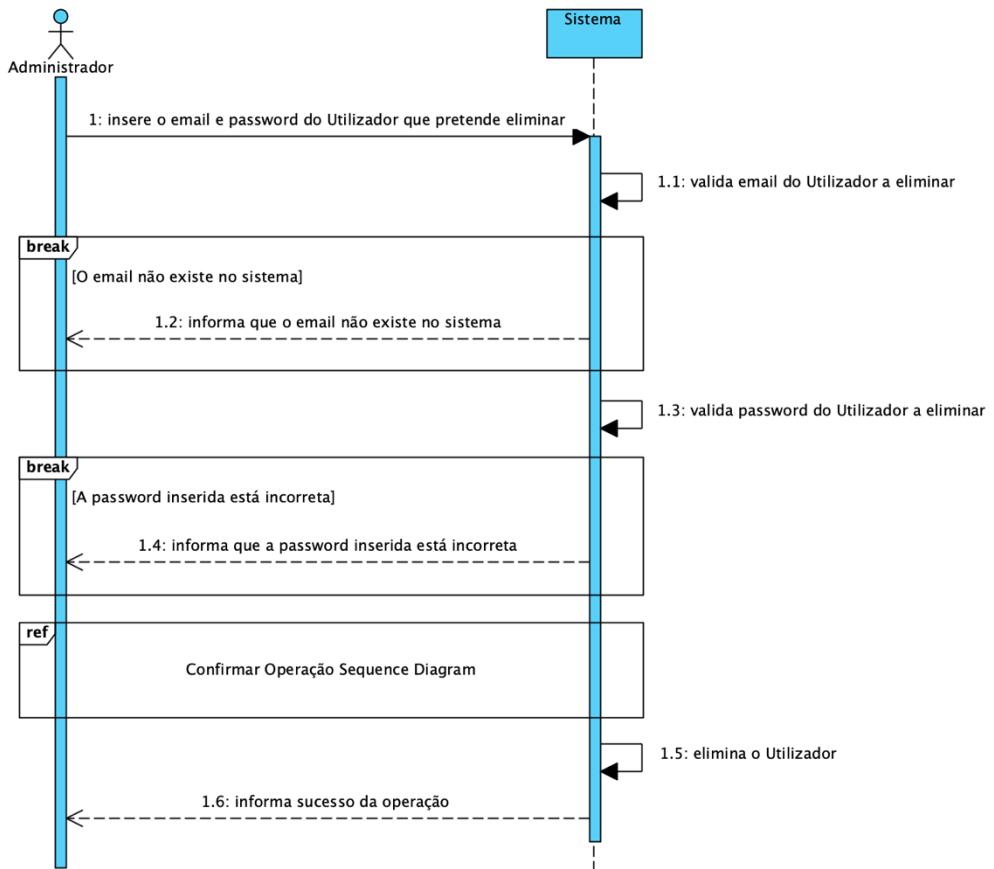
**sd Criar Playlist Pessoal Sequence Diagram**



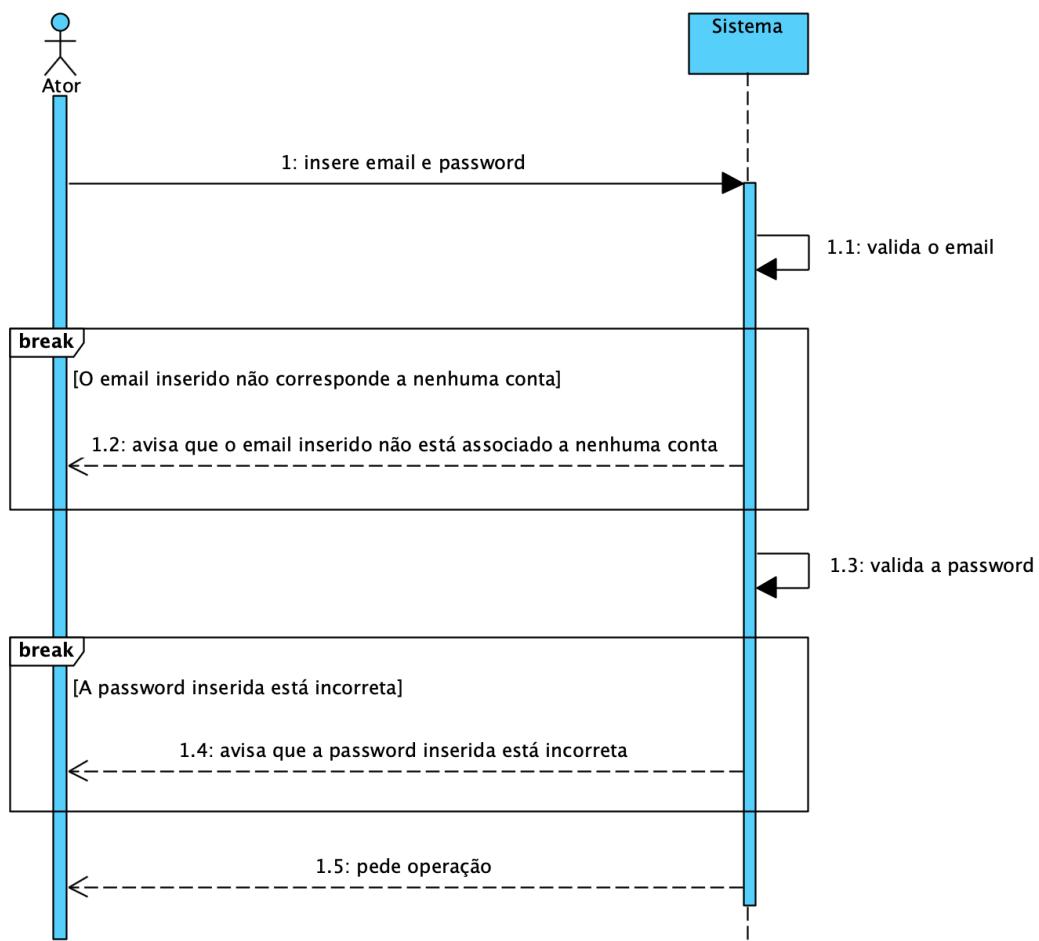
sd Editar Utilizador Sequence Diagram



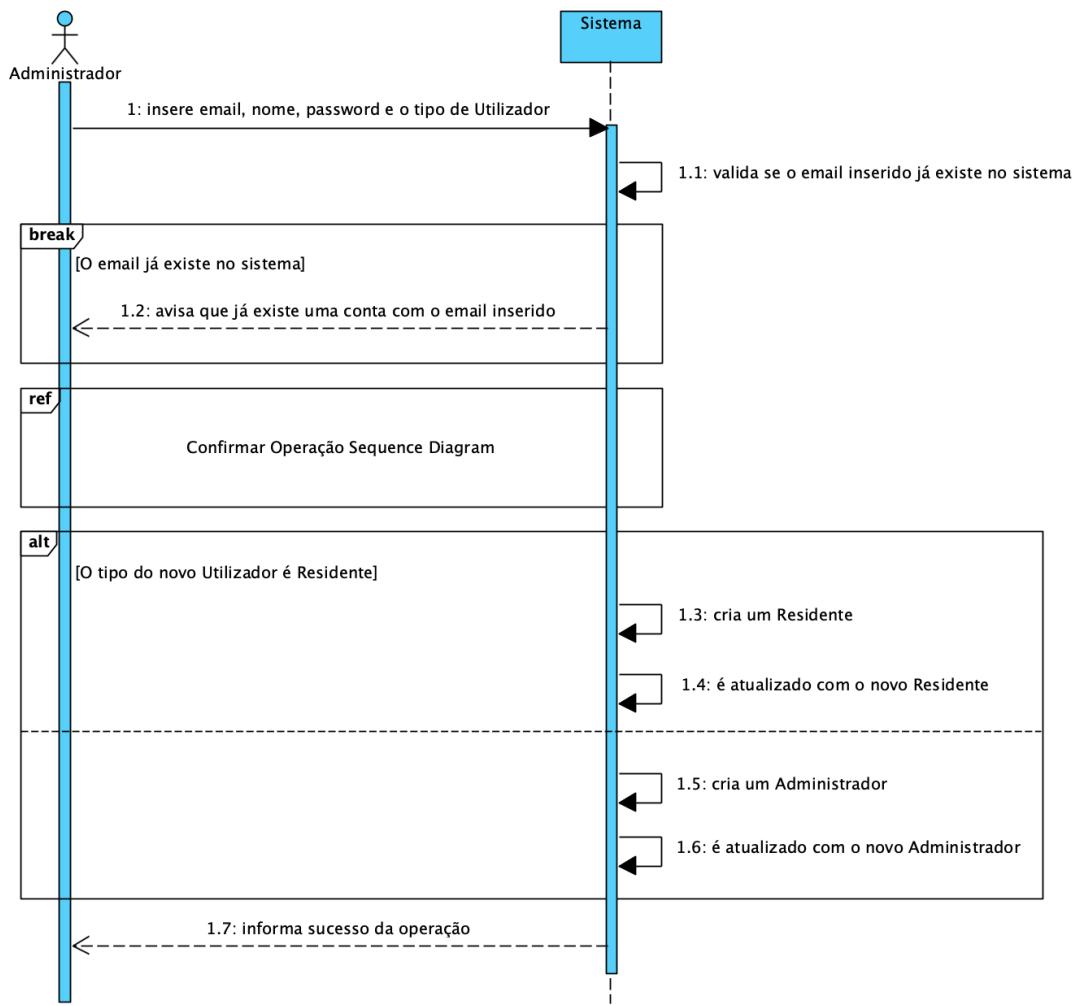
**sd Eliminar Utilizador Sequence Diagram**



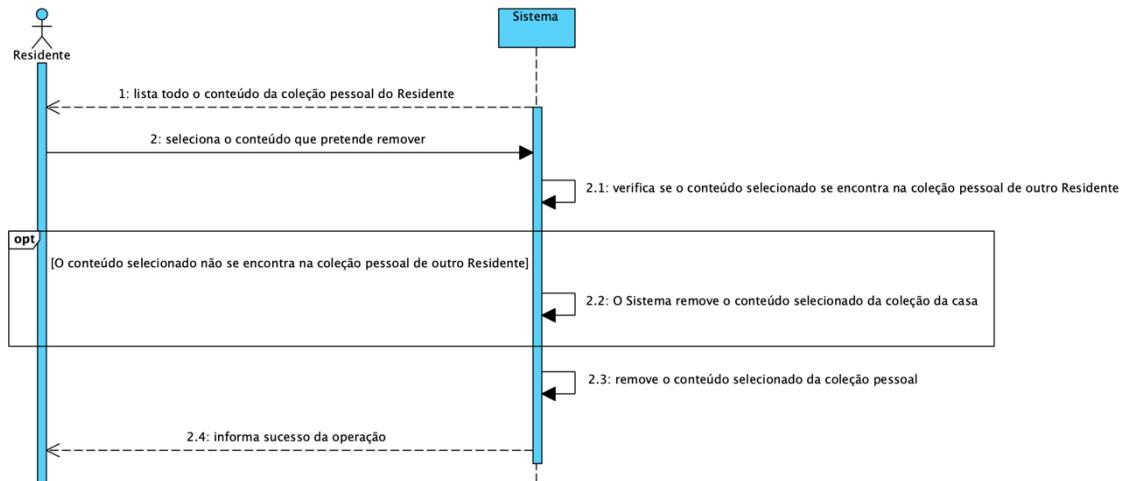
### sd Iniciar Sessão Sequence Diagram



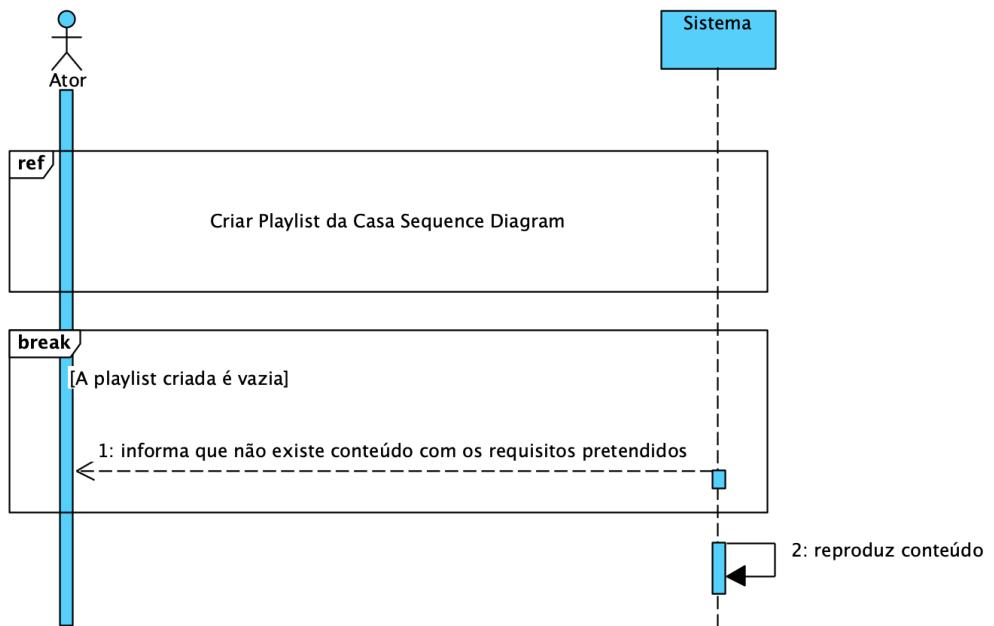
**sd Registar Utilizador Sequence Diagram**



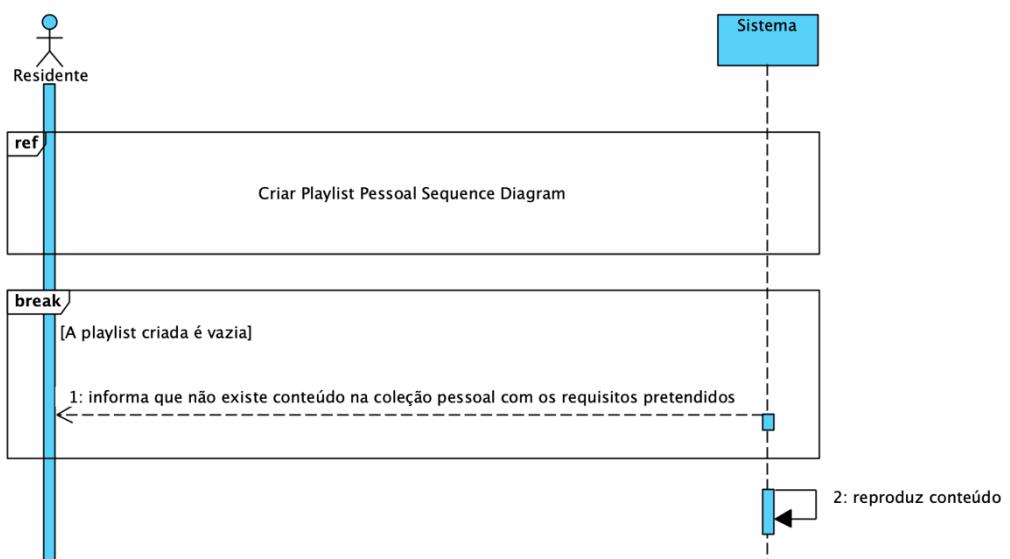
**sd Remover Conteúdo Sequence Diagram**



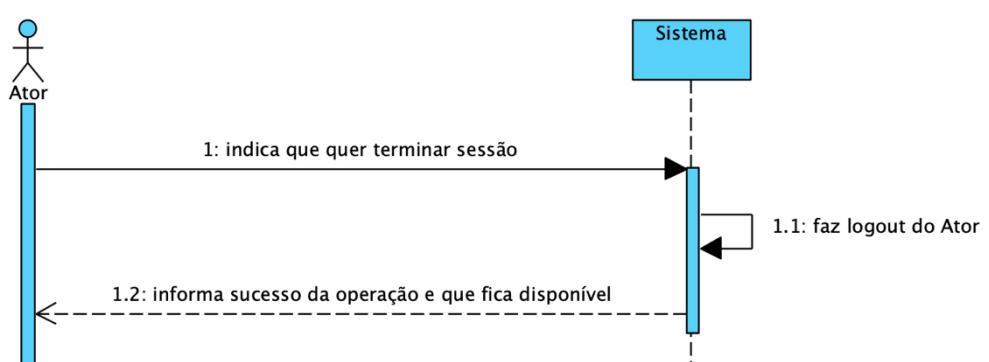
### **sd Reproduzir Conteúdo da Casa Sequence Diagram**



### **sd Reproduzir Conteúdo Pessoal Sequence Diagram**



### **sd Terminar Sessão Sequence Diagram**



## Diagrama de packages

À medida que o software se torna mais complexo, há necessidade de agrupar as classes (packages), que correspondem à abstração de conceitos existentes na linguagem de modelação UML.

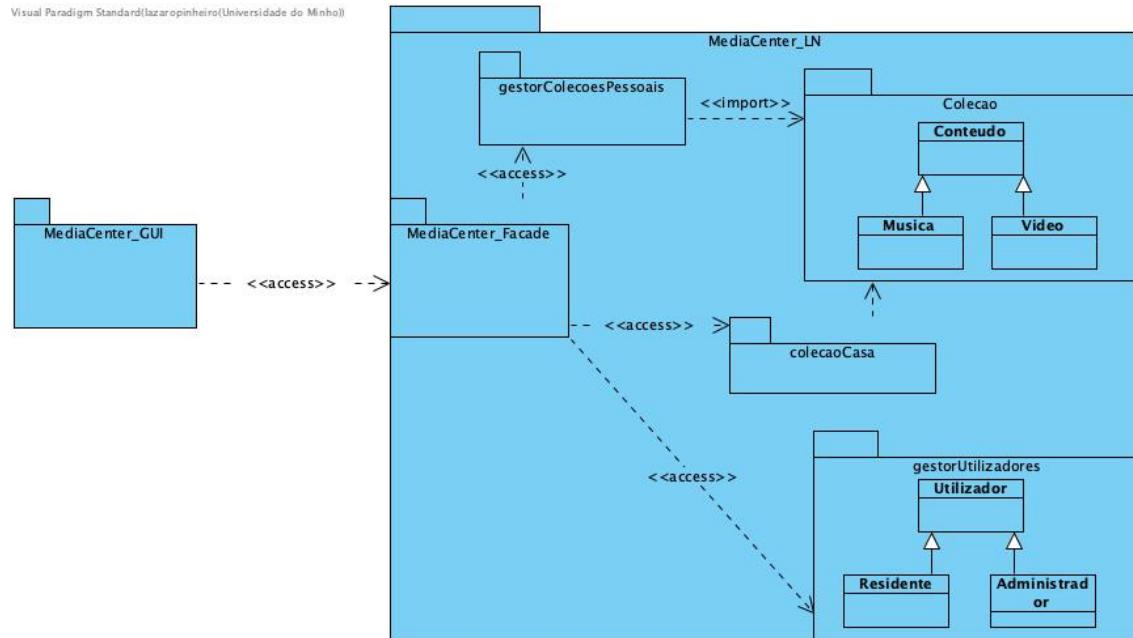


Figura 8 - Diagrama de Packages do media center.

## Diagrama de Classe

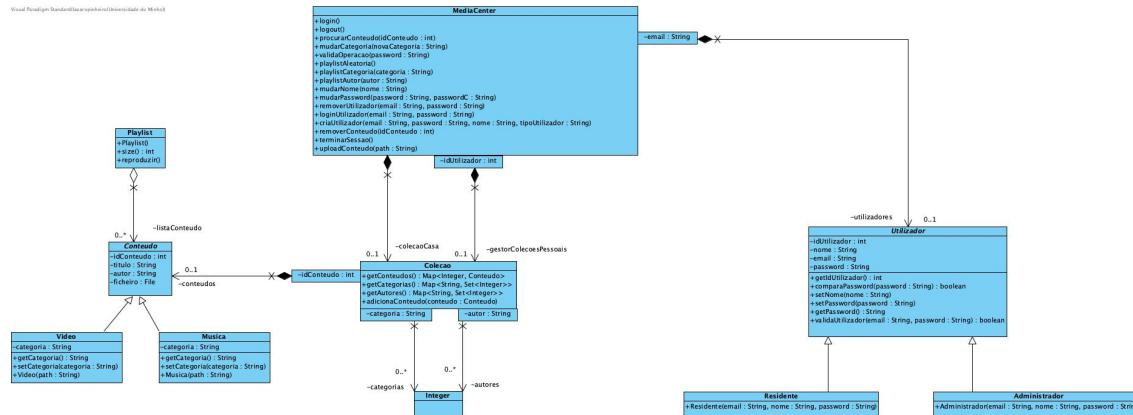


Figura 9 - Diagrama de Classe.

## Diagrama de Sequência das Operações de cada subsistema

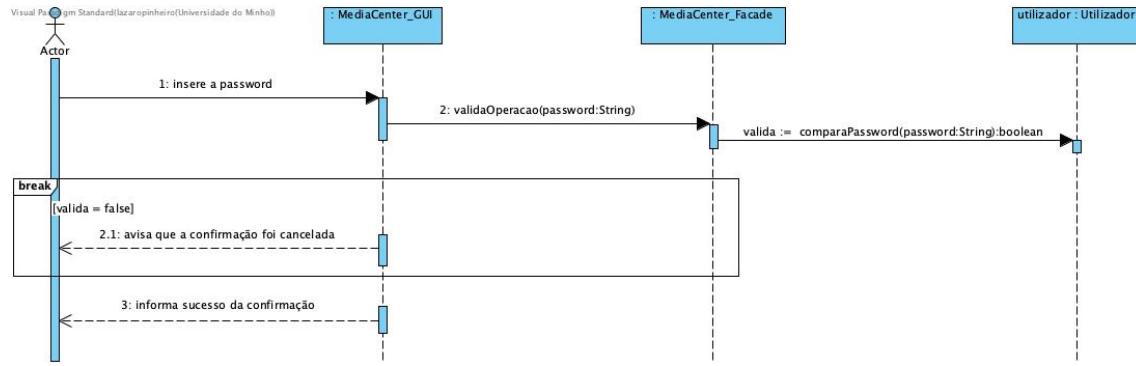


Figura 10 - Diagrama de Sequência estendido de Confirmar Operação

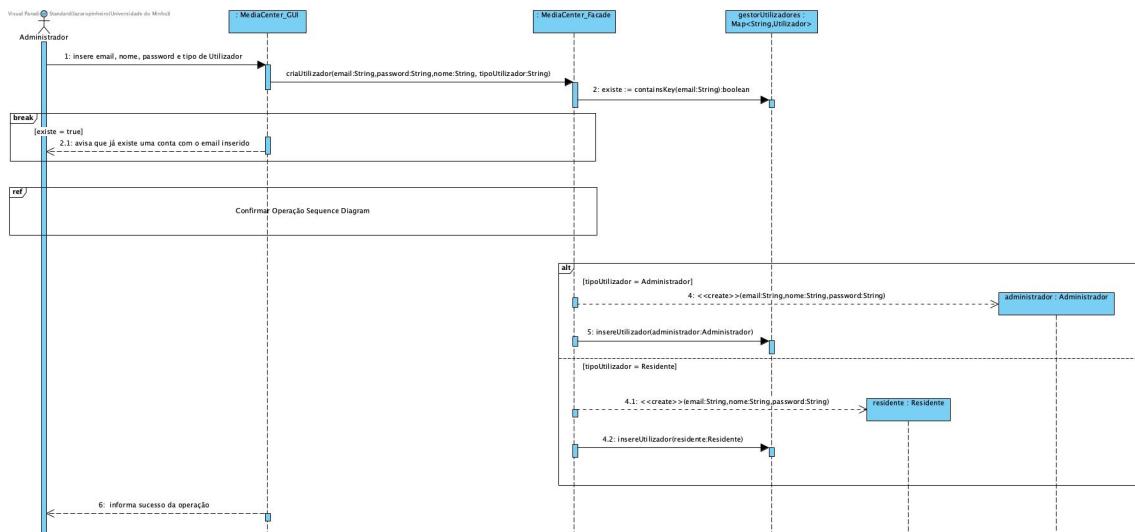


Figura 11 - Diagrama de Sequência estendido de Registar Utilizador

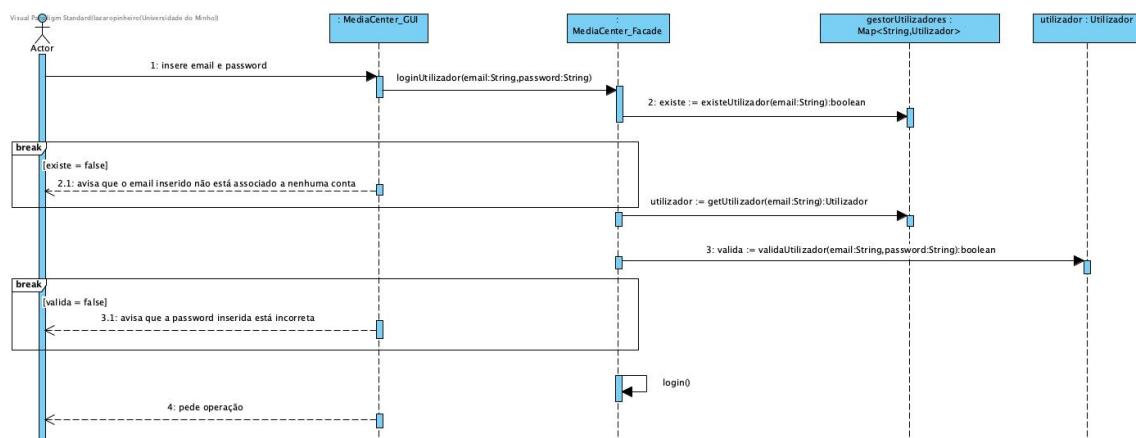


Figura 12 - Diagrama de Sequência estendido de Iniciar Sessão

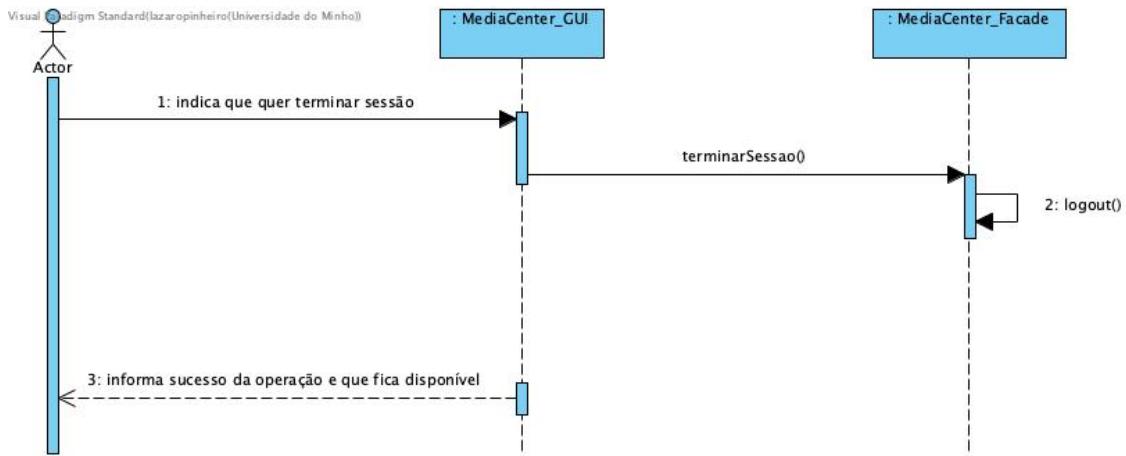


Figura 13 - Diagrama de Sequência estendido de Terminar Sessão

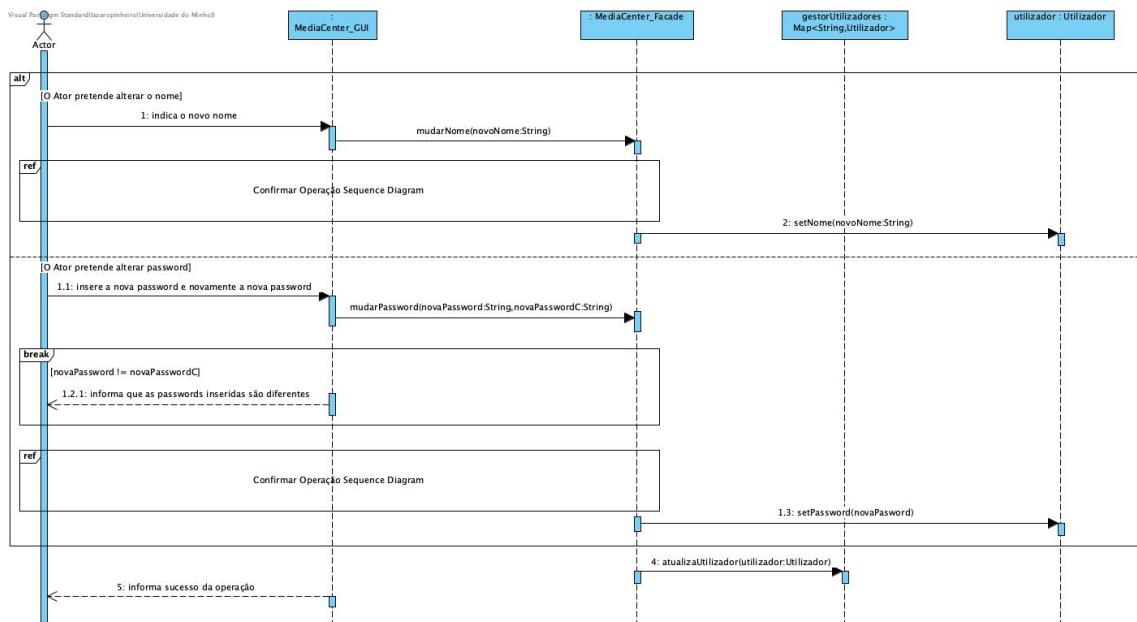


Figura 14 - Diagrama de Sequência estendido de Editar Utilizador

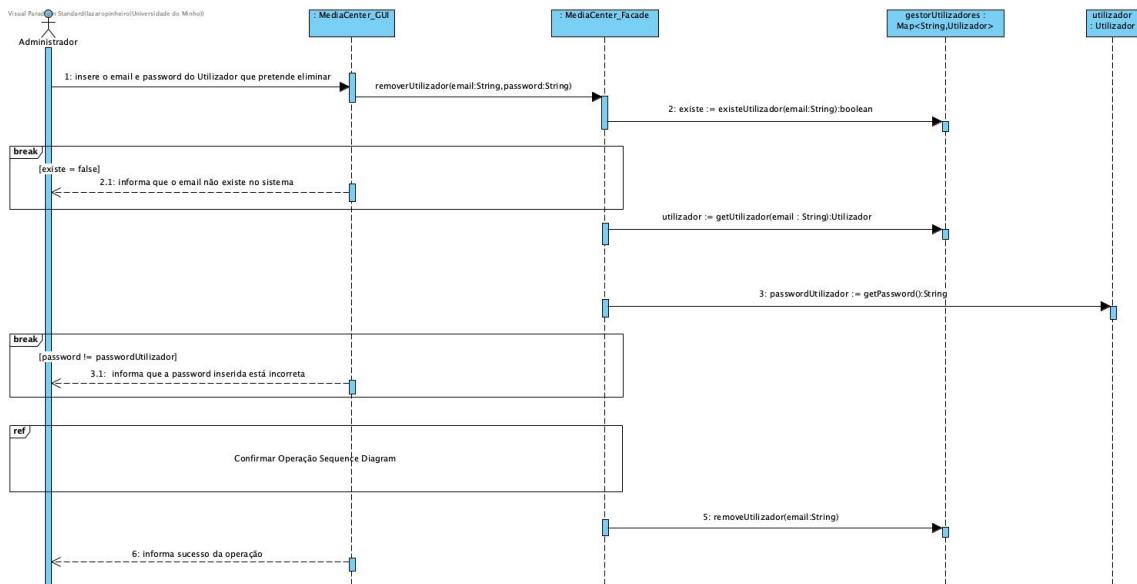


Figura 15 - Diagrama de Sequência estendido de Eliminar Utilizador

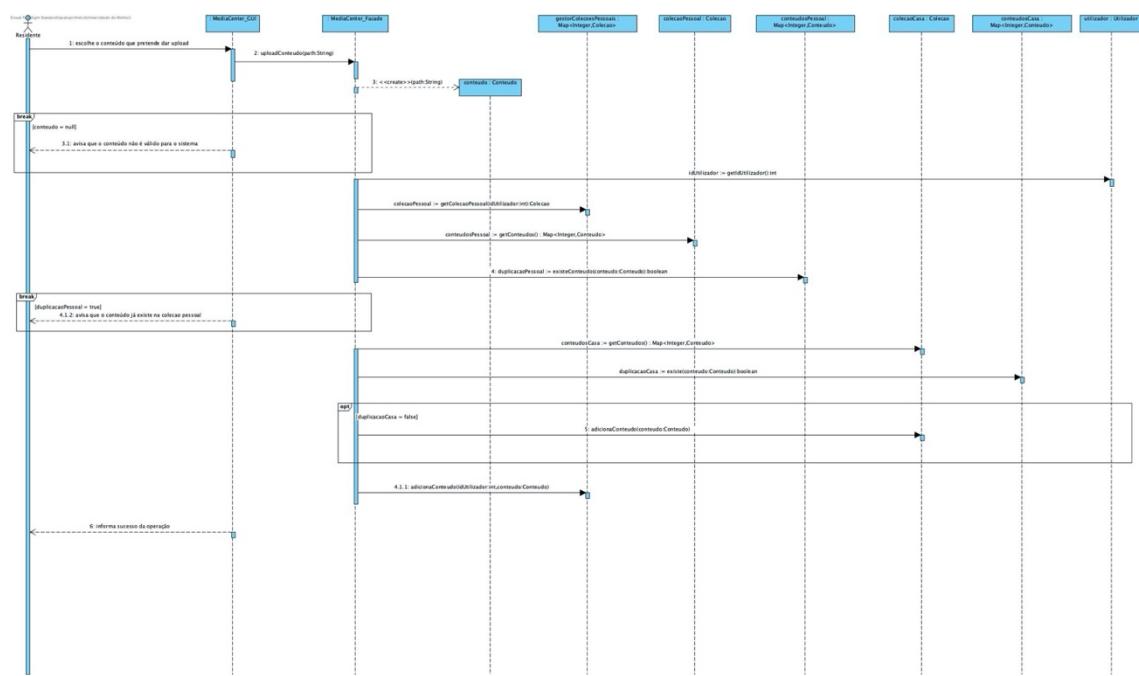


Figura 16 - Diagrama de Sequência estendido de Upload de Conteúdo

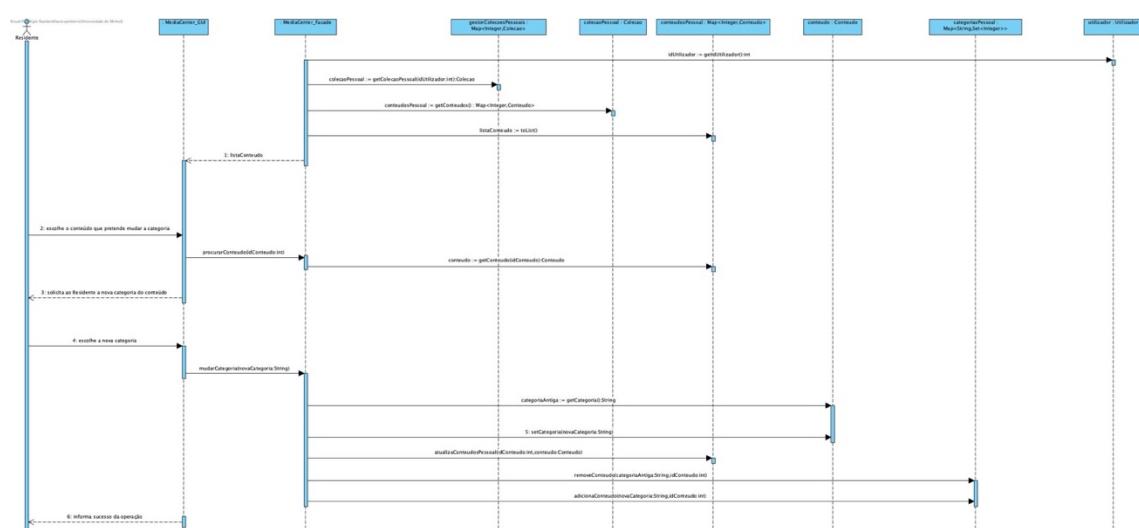


Figura 17 - Diagrama de Sequência estendido de Alterar Categoria de Conteúdo

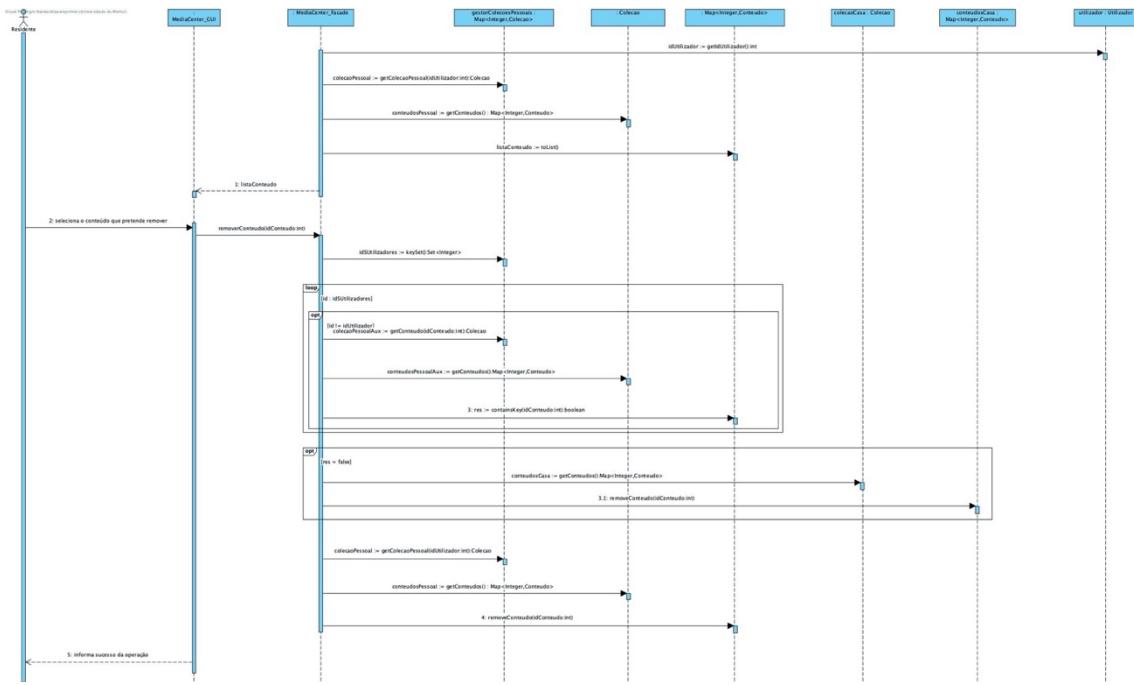


Figura 18 - Diagrama de Sequência estendido de Remover Conteúdo

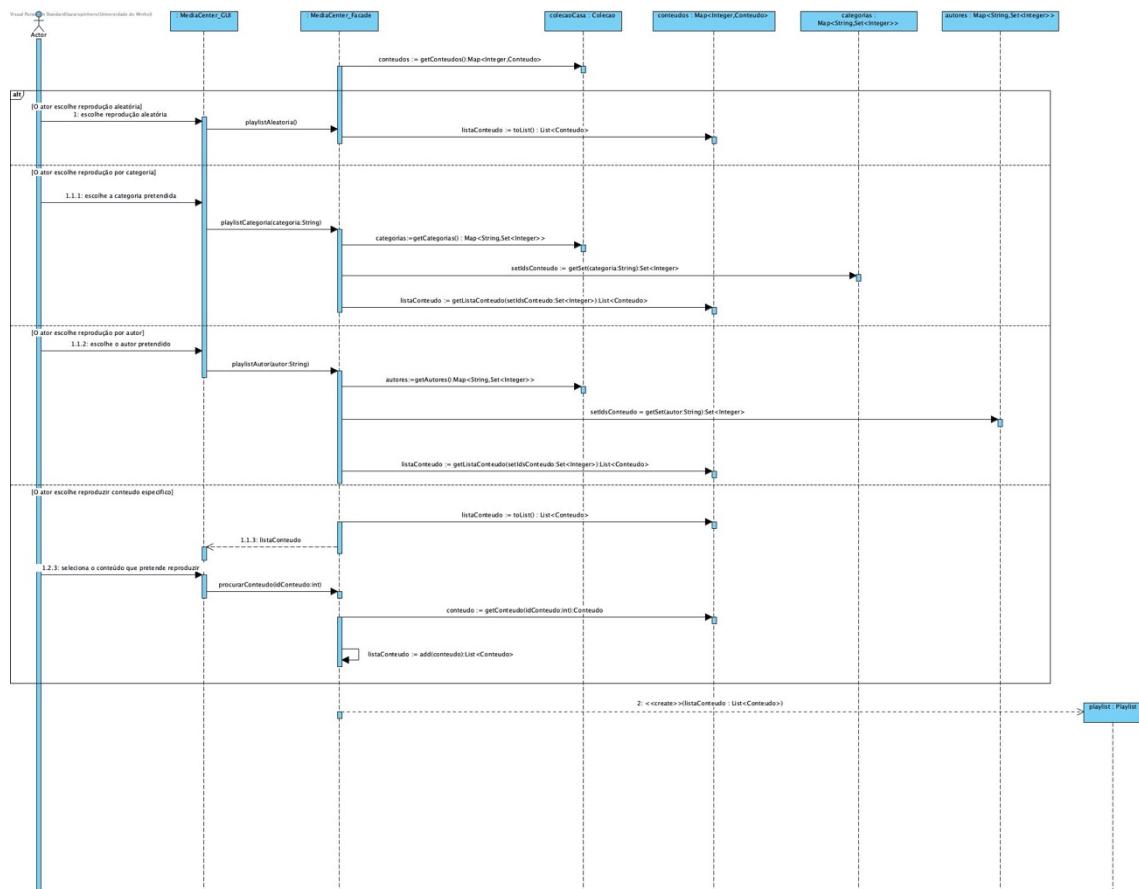


Figura 19 - Diagrama de Sequência estendido de Criar Playlist da Casa

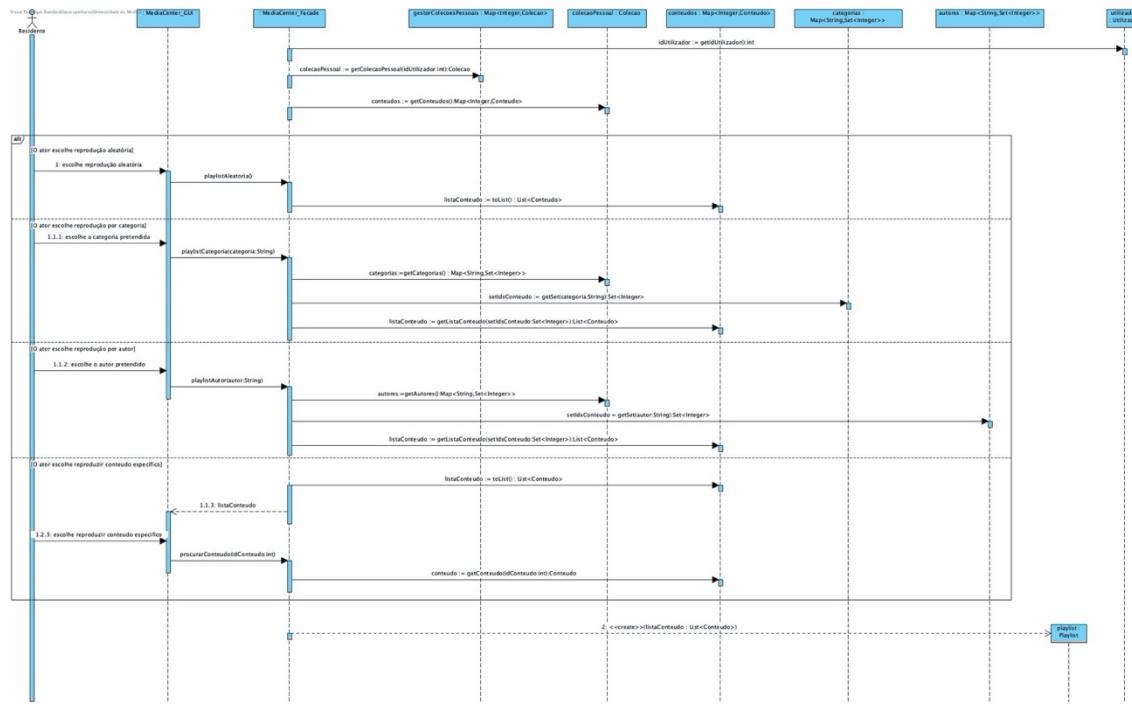


Figura 20 - Diagrama de Sequência estendido de Criar Playlist Pessoal

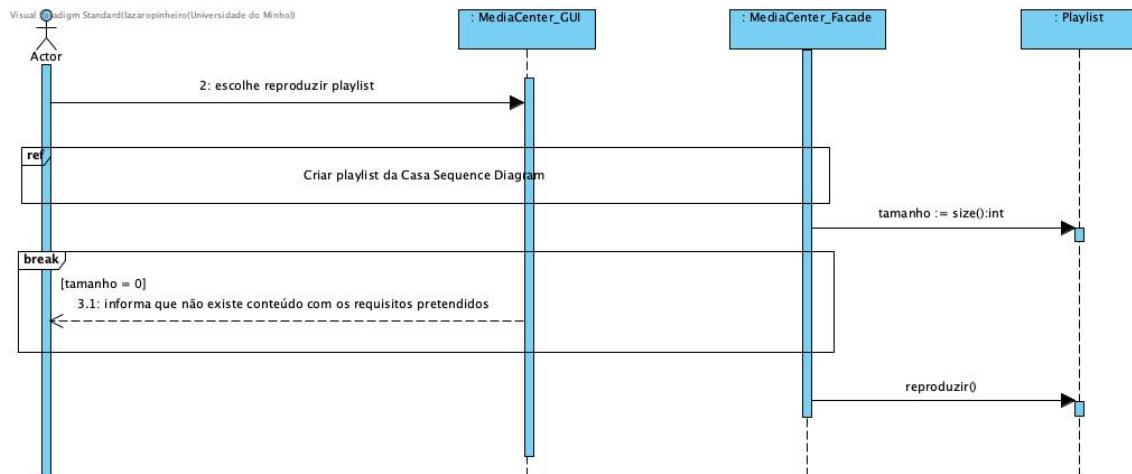


Figura 21 - Diagrama de Sequência estendido de Reproduzir Conteúdo da Casa

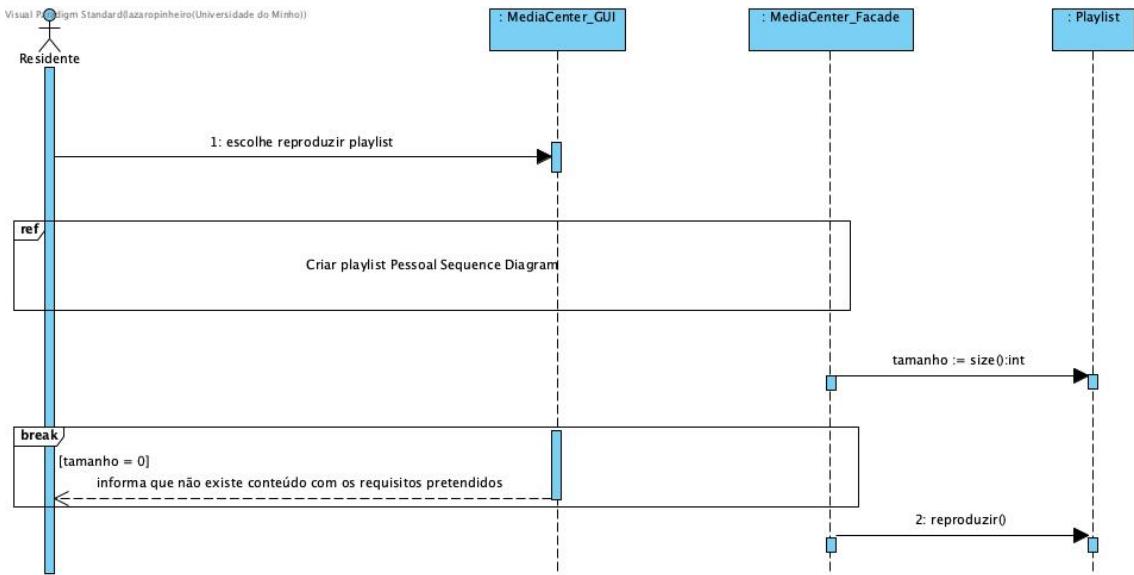
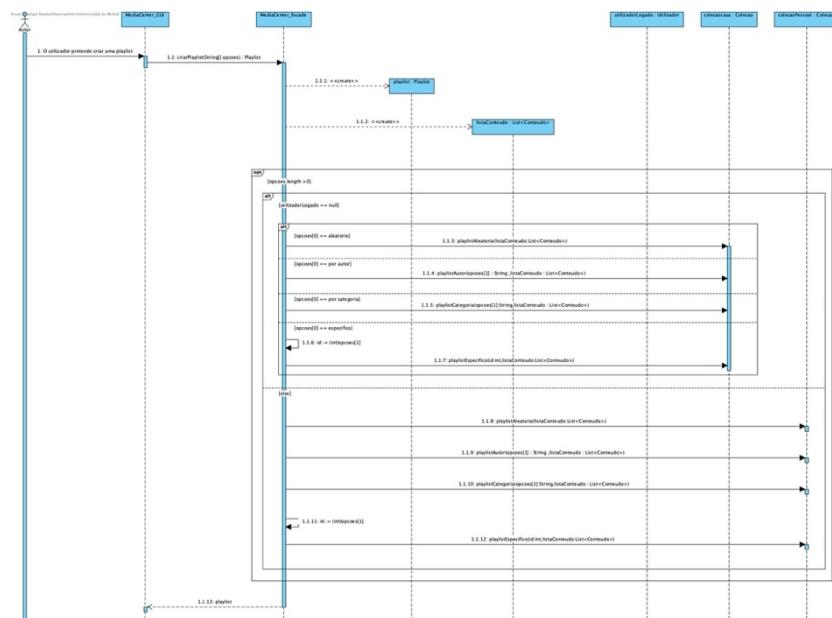


Figura 22 - Diagrama de Sequência estendido de Reproduzir Conteúdo Pessoal

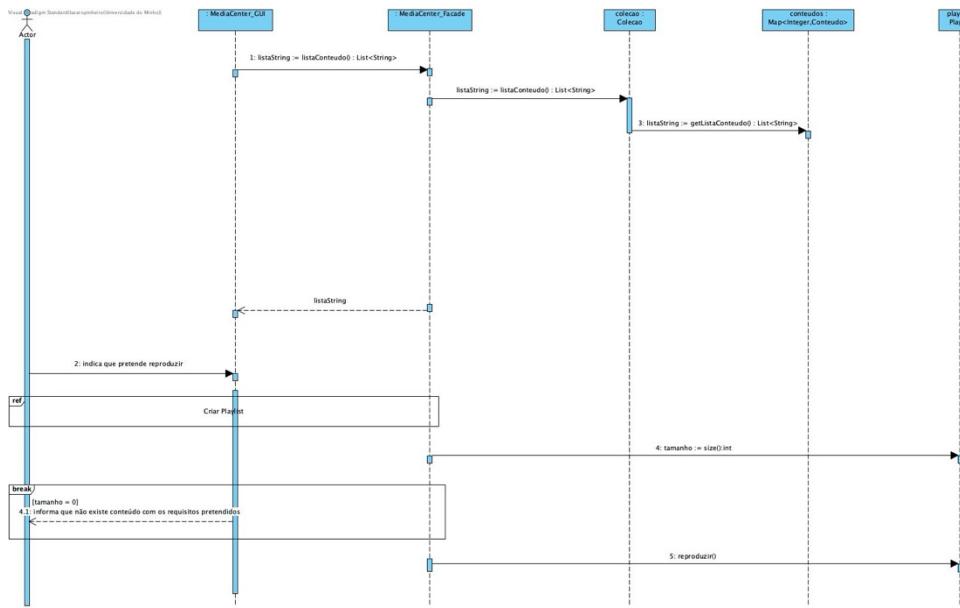
# Implementação

(Considerando os 5 use case solicitados na Fase 2)

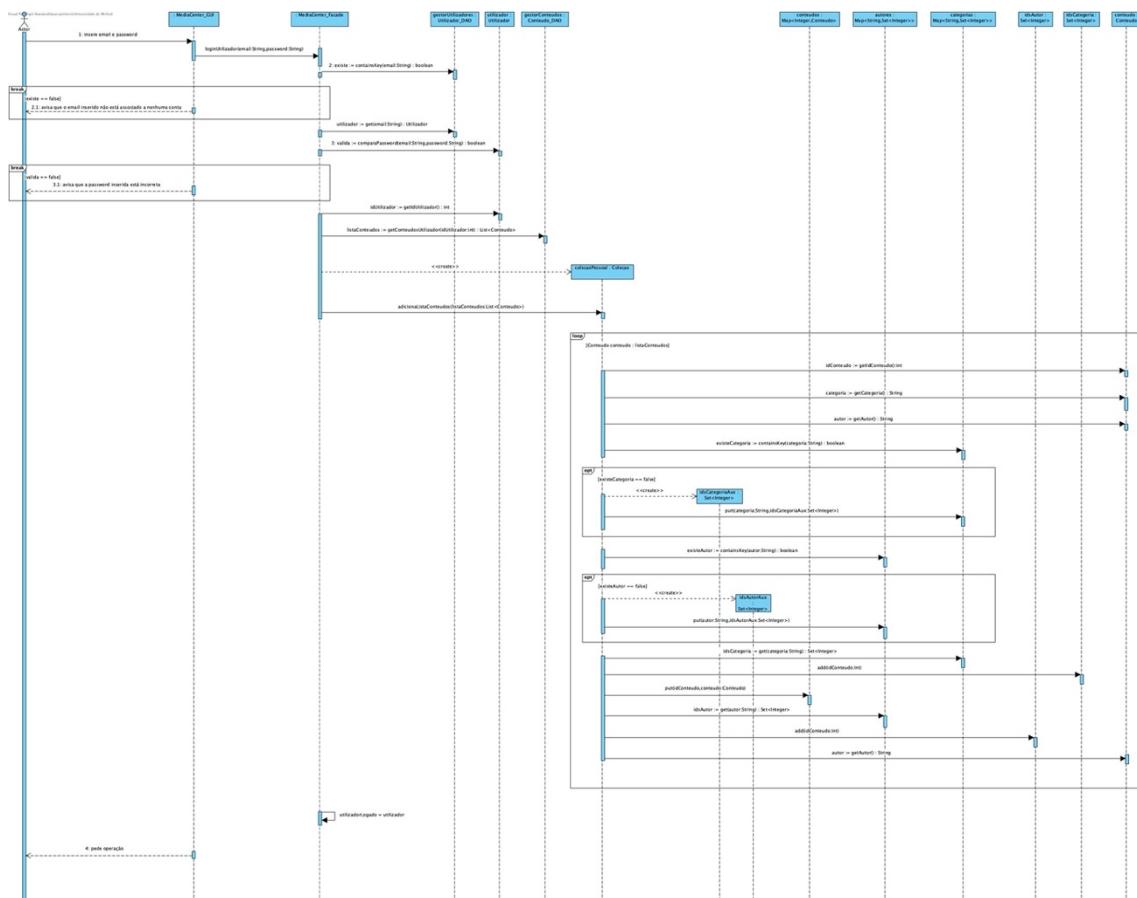
### *Diagrama de Sequência das Operações de cada subsistema*



*Figura 23 - Diagrama Estendido de Criar Playlist*



*Figura 24 - Diagrama Estendido de Reproduzir Conteúdo*



*Figura 25 - Diagrama Estendido de Iniciar Sessão*

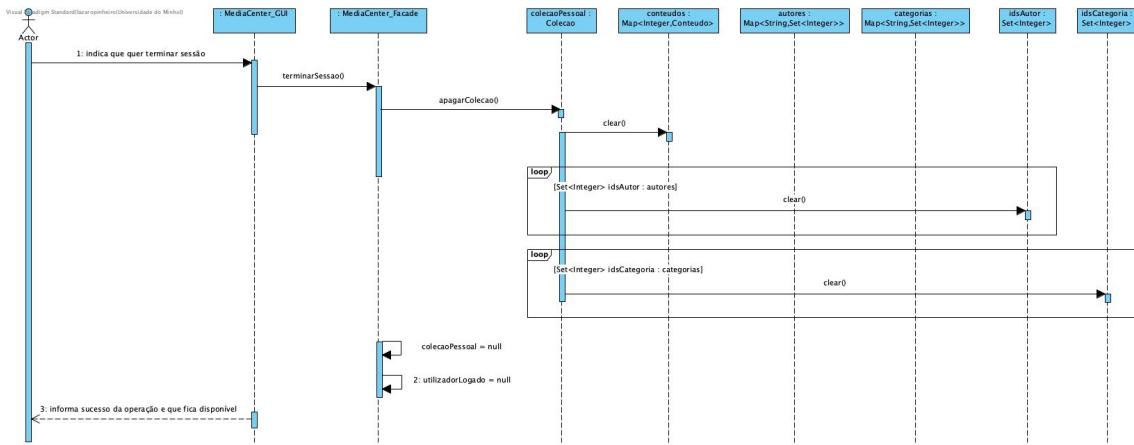


Figura 26 - Diagrama Estendido de Terminar Sessão

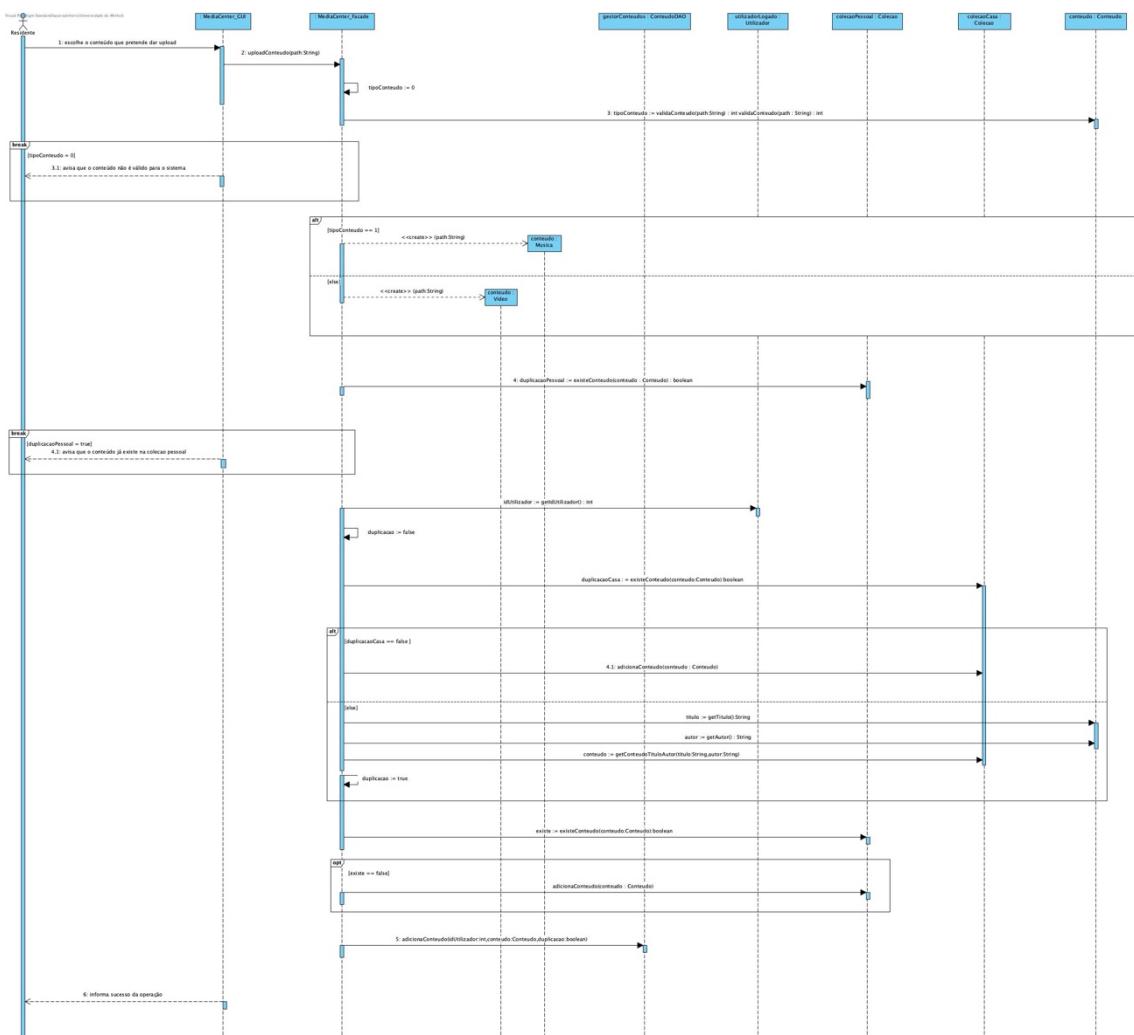
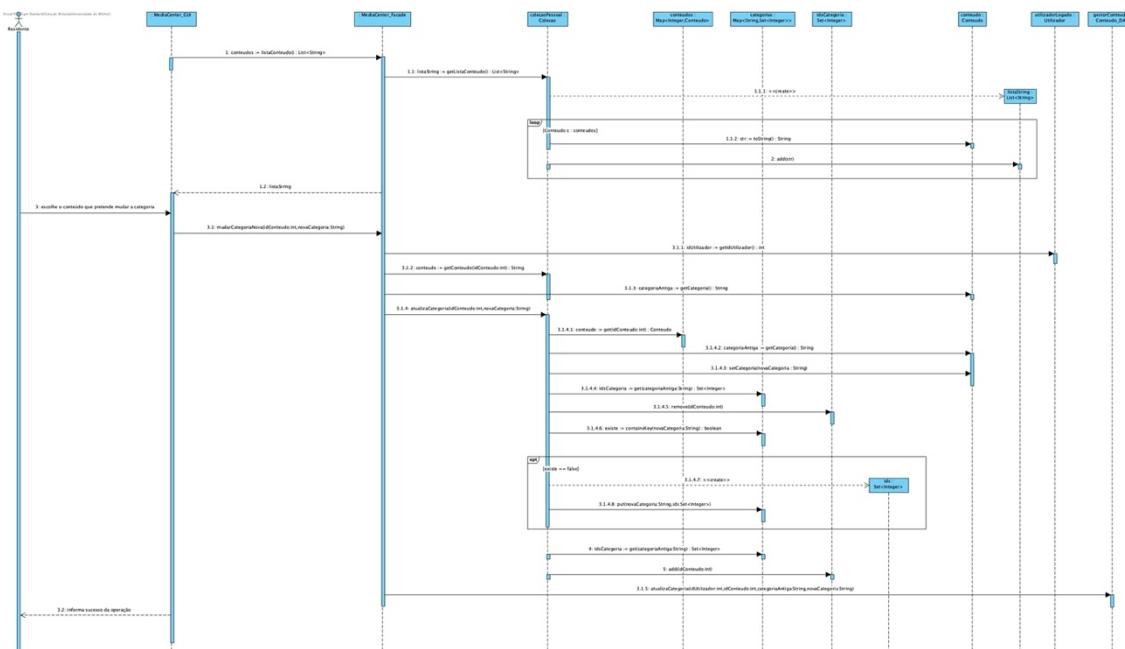
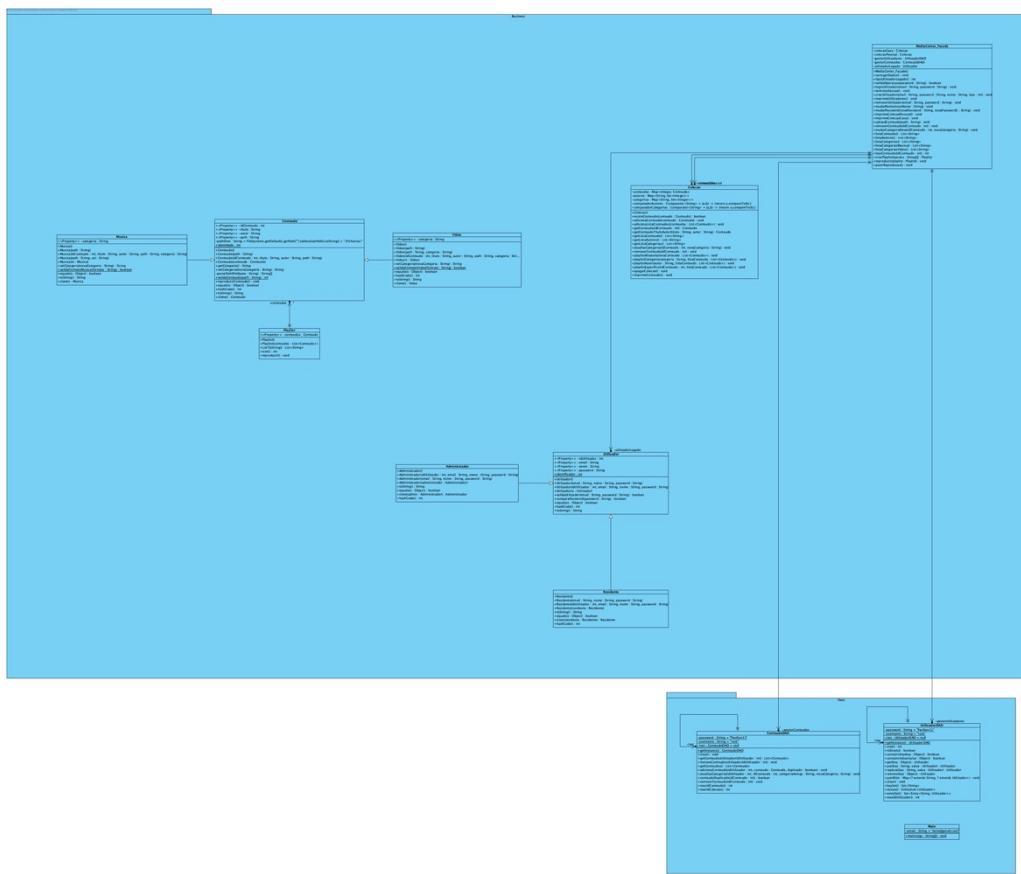


Figura 27 - Diagrama Estendido de Upload de Conteúdo



*Figura 28 - Diagrama Estendido de Alterar Categoria*

Diagrama de Classe com DAO's tabelas do Modelo Relacional



*Figura 29 - Diagrama de Classe usando DAO's*

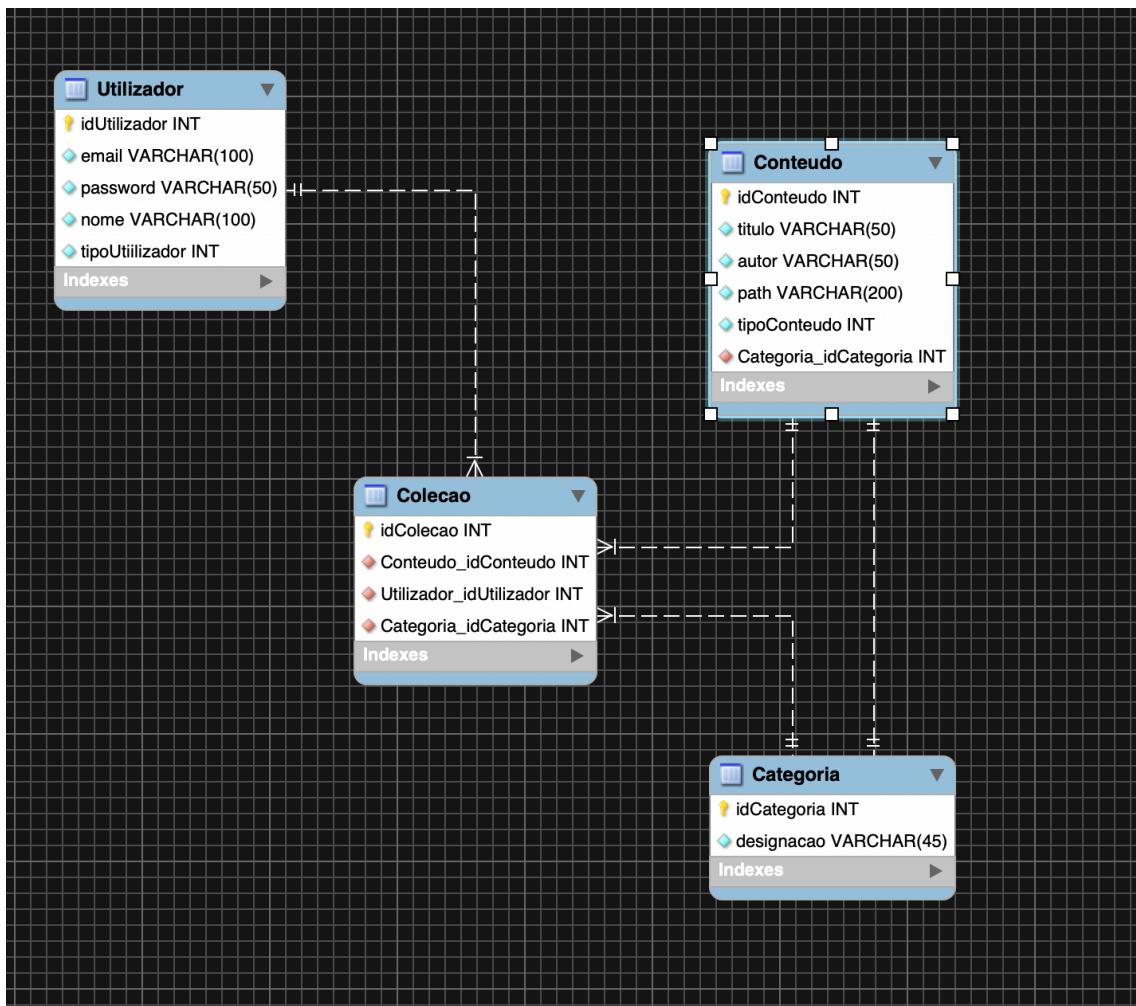


Figura 30 - Tabelas do Modelo Relacional

#### Detalhes relevantes da implementação

Geralmente o comportamento humano, aquando da utilização de um *media center* revela uma maior tendência para efetuar reproduções de conteúdos em detrimento da realização de upload's. Assim sendo, como forma de aumentar a performance, implementamos estruturas de dados dentro do sistema para evitar aceder à base de dados diminuindo os tempos de resposta.

# Conclusão

Numa primeira abordagem, começamos por fazer uma análise mais abstrata daquela que seria a fase final do trabalho. O nosso ponto de partida foi o modelo de domínio, que nos ajudou a ter uma visão mais clara sobre aquilo que pretendíamos da aplicação. Seguiu-se o desenvolvimento dos Use Cases aliado a uma primeira interface onde refletimos sobre como seria a interação entre os atores e o sistema da aplicação. Esta primeira fase do trabalho permitiu-nos formular melhor o futuro do trabalho. Como a nossa primeira abordagem foi muito simples, na segunda fase do trabalho, adicionamos mais alguns use cases e repensamos alguns deles, segundo as indicações do professor. Esta foi uma fase mais trabalhosa tendo em conta a construção dos diagramas de sequência estendidos. Construímos o diagrama de classes partindo do modelo de domínio e, à medida que os diagramas de sequência estendidos eram realizados atualizávamos o nosso diagrama de classes, com os métodos e atributos que pertenciam a cada classe.

Na fase de implementação do trabalho já possuímos uma base estruturada, o que tornou mais simples a implementação. O facto de o trabalho estar dividido em 3 fases facilitou-nos a construção da fase seguinte, embora tenha sido um processo longo e trabalhoso. Nesta última fase foram atualizados os diagramas de sequência estendidos e o diagrama de classe tendo em conta os DAO's.

Em conclusão, a aplicação construída foi bem conseguida e é de fácil utilização, cumprindo todos os requisitos propostos no enunciado do trabalho e as instruções do professor. A modelagem foi feita com a maior das atenção e da melhor forma que conseguimos, servindo de base para a implementação. Temos então uma aplicação que poderia ser explorado e utilizado no dia a dia.

# Anexos

Anexo 1

## **Confirmar Operação**

Descrição: O Sistema solicita a password do Ator para confirmar operação

Ator: Administrador/Residente

Cenários:

Pré-condição: O Ator está autenticado

Pós-condição: A operação é confirmada

Fluxo normal:

1 – O Ator insere a sua password

2 – O Sistema valida password do Ator

3 – O Sistema informa sucesso da operação

Fluxo de exceção 1:

[A password do Ator está incorreta] (passo 2)

2.1 – O sistema avisa que a operação foi cancelada

## **Registrar Utilizador**

Descrição: O Administrador cria um novo Utilizador no sistema

Ator: Administrador

Cenários: “A Isabel, pediu à Paula, que tinha permissões de administração, que criasse a conta do Manuel no *media center*.”

Pré-condição: O Administrador está autenticado

Pós-condição: O Sistema fica atualizado com novo Utilizador

Fluxo normal:

- 1 – O Administrador insere email, nome, password e o tipo de Utilizador do novo Utilizador
- 2 - <<include>> Confirmar Operação
- 3 - O Sistema valida se o email inserido já existe no Sistema
- 4 – O Sistema cria um Administrador
- 5 – O Sistema é atualizado com o novo Administrador
- 6 – O Sistema informa sucesso da operação

Fluxo alternativo:

- [O tipo do novo Utilizador é Residente] (passo 4)
- 4.1 – O Sistema cria um Residente
  - 4.2 – O Sistema é atualizado com o novo Residente
  - 4.2 – Volta ao passo 6

Fluxo de exceção 1:

- [O email já existe no sistema] (passo 2)
- 2.1 – O sistema avisa que já existe uma conta com o email inserido

## **Iniciar Sessão**

Descrição: O Ator autentica-se no sistema

Ator: Administrador/Residente

Cenários:

Pré-condição: O Ator tem conta e nenhum Ator autenticado.

Pós-condição: O Ator fica autenticado no sistema

Fluxo normal:

- 1 – O Ator insere email e password
- 2 – O Sistema valida o email
- 3 – O Sistema valida a password
- 4 – O Sistema pede operação

Fluxo de exceção 1:

[O email inserido não corresponde a nenhuma conta] (passo 2)

2.1 – O sistema avisa que o email inserido não está associado a nenhuma conta

Fluxo de exceção 2:

[A password inserida está incorreta] (passo 3)

3.1 – O sistema avisa que a password inserida está incorreta

## **Terminar Sessão**

Descrição: O Ator termina a sessão

Autor: Administrador/Residente.

Cenários: “..., o Ricardo fez *logout* do *media center* ...”

Pré-condição: O Ator está autenticado no sistema.

Pós-condição: O Sistema fica sem ninguém autenticado.

Fluxo normal:

1 – O Ator indica que quer terminar sessão

2 – O Sistema faz *logout* do Ator

3 – O Sistema informa sucesso da operação e que fica disponível

## **Editar Utilizador**

Descrição: O Ator edita os seus atributos

Ator: Residente/Administrador

Cenários:

Pré-condição: O Ator está autenticado no sistema

Pós-condição: O Sistema é atualizado

Fluxo normal:

- 1 – O Ator indica o novo nome
- 2 – <<include>> Confirmar Operação
- 3 – O Sistema atualiza o nome do Utilizador
- 4 – O Sistema informa sucesso da operação

Fluxo alternativo 1:

- [O Ator pretende alterar password] (passo 1)
- 1.1 – O Ator insere a nova password e novamente a nova password
  - 1.2 - <<include>> Confirmar Operação
  - 1.3– O Sistema valida as passwords
  - 1.4 – O Sistema atualiza a password do Utilizador
  - 1.5 - Volta ao passo 4

Fluxo de exceção 1:

- [Passwords são diferentes] (passo 1.3)
- 1.3.1 – O sistema informa que as passwords inseridas são diferentes

## **Eliminar Utilizador**

Descrição: O Administrador remove um Utilizador do sistema

Ator: Administrador

Cenários:

Pré-condição: O Administrador está autenticado no sistema

Pós-condição: O Sistema fica atualizado e o Utilizador é removido do Sistema

Fluxo normal:

1 – O Administrador insere o email e password do Utilizador que pretende eliminar

2 – <<include>> Confirmar Operação

3 – O Sistema valida email do Utilizador a eliminar

4 – O Sistema valida password do Utilizador a eliminar

5 – O Sistema elimina o Utilizador

6 – O Sistema informa sucesso da operação

Fluxo de exceção 1:

[O email não existe no sistema] (passo 3)

3.1 – O Sistema informa que o email não existe no sistema

Fluxo de exceção 2:

[A password inserida está incorreta] (passo 4)

4.1 – O sistema informa que a password inserida está incorreta

## **Upload de Conteúdo**

Descrição: O Residente dá upload de conteúdo para o sistema

Ator: Residente

Cenários: "..., em que poderia fazer upload da sua música e vídeos, que passariam a estar disponíveis para todos os residentes, ... ", "..., dirigindo-se depois à sala e utilizou-o para fazer upload do conteúdo para a sua conta no media center."

Pré-condição: O Residente está autenticado

Pós-condição: O conteúdo fica disponível no sistema para reprodução

Fluxo normal:

- 1 – O Residente escolhe o conteúdo que pretende dar upload
- 2 - O Sistema carrega o conteúdo selecionado
- 3 – O Sistema valida ficheiro do conteúdo
- 4 – O Sistema verifica duplicação do conteúdo
- 5 – O sistema disponibiliza o conteúdo carregado na coleção da casa e na coleção pessoal
- 6 – O Sistema informa sucesso da operação

Fluxo alternativo 1:

[O conteúdo já existe na coleção da casa] (passo 4)

4.1 – O Sistema disponibiliza o conteúdo na coleção pessoal

4.2– Volta ao passo 6

Fluxo de exceção 1:

[O ficheiro do conteúdo não é válido para o sistema] (passo 3)

3.1 – O sistema avisa que o conteúdo não é válido para o sistema

Fluxo de exceção 2:

[O conteúdo já existe na coleção pessoal] (passo 4)

4.1 – O sistema avisa que o conteúdo já existe na coleção pessoal

## **Alterar Categoria de Conteúdo**

Descrição: O Residente altera a categoria do conteúdo na sua coleção pessoal

Ator: Residente

Cenários: “O Manuel notou que o sistema gerara automaticamente um conjunto de categorias para organizar o conteúdo (com base na informação já existente no sistema), mas não gostou da forma como alguma da música tinha ficado classificada, pelo que gastou algum tempo a organizar o conteúdo de acordo com a sua preferência.”

Pré-condição: O Residente está autenticado e tem conteúdo na sua coleção pessoal

Pós-condição: O conteúdo da coleção pessoal do Residente é alterado

Fluxo normal:

- 1 – O Sistema lista todo o conteúdo da coleção pessoal do Residente
- 2 – O Residente escolhe o conteúdo que pretende mudar a categoria
- 3 – O Residente escolhe a nova categoria
- 4 – O Sistema altera a categoria do conteúdo na coleção pessoal
- 5 – O Sistema informa sucesso da operação

## **Remover Conteúdo**

Descrição: O Residente remove conteúdo selecionado

Ator: Residente

Cenários:

Pré-condição: O Residente está autenticado e tem conteúdo na sua coleção pessoal

Pós-condição: O Conteúdo selecionado é removido da coleção pessoal do Residente

Fluxo normal:

- 1 - O Sistema lista todo o conteúdo da coleção pessoal do Residente
- 2 – O Residente seleciona o conteúdo que pretende remover
- 3 – O Sistema verifica se o conteúdo selecionado se encontra na coleção pessoal de outro Residente
- 4 – O sistema remove o conteúdo selecionado da coleção pessoal
- 5 – O Sistema informa sucesso da operação

Fluxo alternativo 1:

[O conteúdo selecionado não se encontra na coleção pessoal de outro Residente] (passo 3)

- 3.1 – O Sistema remove o conteúdo selecionado da coleção da casa
- 3.2 – Volta ao passo 4

## **Criar Playlist**

Descrição: O Utilizador cria uma playlist de reprodução

Ator: Convidado/Residente/Administrador

Cenários:

Pré-condição: O sistema tem conteúdo

Pós-condição: O sistema cria uma playlist de reprodução

Fluxo normal:

1 – O utilizador seleciona qual o tipo de reprodução

2 – O sistema gera uma playlist com base na opção do utilizador e no seu estado relativamente ao sistema.

3 – O sistema devolve a playlist gerada

## **Reproduzir Conteúdo**

Descrição: O Residente reproduz uma playlist

Ator: Residente/Convidado

Cenários: “Colocou-o a tocar Jazz em modo aleatório, mas apenas com base na música presente na sua coleção.”

Pré-condição: O Ator está autenticado e o sistema tem conteúdo na coleção pessoal

Pós-condição: O sistema reproduz conteúdo

Fluxo normal:

1 – O Ator escolhe reproduzir playlist

2 – <<include>> Criar Playlist

3 – O sistema reproduz conteúdo

Fluxo de exceção 1:

[A playlist criada é vazia] (passo 2)

**2.1 – O Sistema informa que não existe conteúdo na coleção pessoal com os requisitos pretendidos**