

Netflix

Programação Orientada pelos Objectos
Enunciado do 2º Trabalho Prático, versão 1.4 – 2018-05-15
Contacto: carla.ferreira@fct.unl.pt

Notas importantes

Prazo de entrega até às 23h59 do dia 1 de Junho de 2018.

Material a entregar neste trabalho REALIZADO EM GRUPOS DE 2 ALUNOS

- **Mooshak:** submissão e aceitação do código fonte pelo sistema de avaliação automática Mooshak (pontuação > 0). Consulte na página da disciplina as regras de submissão de trabalhos ao Mooshak.
- Incluir no código submetido ao Mooshak:
 - a definição e verificação de **invariantes relacionados com o perfil *Criança*** através do **mecanismo de asserções** do Java.
 - **testes unitários** que verifiquem o correcto funcionamento da **lista dos programas vistos recentemente**. Ou seja, devem desenvolver testes unitários, usando a Framework JUnit, para os métodos da interface que representa um perfil relacionados com a lista de programas vistos recentemente.

Recomendações: A boa documentação do seu código fonte será valorizada, tal como a utilização do melhor estilo de programação possível, para além, claro, do correcto funcionamento do projecto. Não se esqueça de comentar as declarações das interfaces e das classes, bem como dos seus métodos, com uma explicação do seu significado, colocando pré-condições sempre que se justifique. Os alunos podem e devem tirar todas as dúvidas com os docentes da disciplina, bem como conversar com os colegas e debater soluções – mas nunca partilhando código – sempre respeitando os princípios do Código de Ética (disponível na página da cadeira no Moodle).

1 Desenvolvimento da Aplicação *Netflix*

1.1 Descrição do Problema

O objectivo deste trabalho é o desenvolvimento de uma aplicação para a gestão de um serviço de streaming que permite aos clientes verem uma variedade de filmes e séries de televisão em qualquer dispositivo ligado à internet. Os clientes do serviço de streaming usam a aplicação para aderir, cancelar, ou modificar o plano de adesão ao serviço¹. A aplicação permite também aos seus membros acesso instantâneo à biblioteca global de conteúdos com filmes e séries de televisão.

¹Não será considerado o cancelamento de contas

Cada conta é caracterizada pelo nome, email e password do cliente. Para além desta informação, cada conta tem um plano de adesão: *Base*, *Standard* ou *Premium*. O plano de adesão determina quantos dispositivos podem ser conectados ao serviço e o número de perfis personalizáveis que podem ser criados. Qualquer acesso ao serviço de streaming é feito através de um dispositivo. Se o dispositivo é novo este será adicionado aos dispositivos associados à conta, caso contrário o conjunto de dispositivos associados à conta não é modificado. A Tabela 1 apresenta os detalhes dos vários planos disponíveis.

Plano	Número de Dispositivos	Número de Perfis
<i>Basic</i>	1	2
<i>Standard</i>	2	5
<i>Premium</i>	4	5

Tabela 1: Planos de adesão.

Os perfis permitem que diferentes membros da família possam personalizar o serviço de acordo com as suas preferências. Todos os planos permitem criar perfis *Criança* que apresentam exclusivamente conteúdo para crianças de acordo com a classificação etária definida. Por exemplo, se a classificação etária do perfil for 12, só poderão ser vistos programas com classificação etária inferior ou igual a 12. Cada perfil tem um nome e a lista dos programas vistos recentemente (dez últimos programas vistos).

O serviço tem uma vasta biblioteca de conteúdos com filmes e séries de televisão. A informação a manter para os programas é a seguinte:

Filmes. Cada filme tem os seguintes dados associados: o título; o director; a duração; a classificação etária; o ano de lançamento; o género; o elenco. Considere que cada filme é univocamente identificado pelo título.

Séries de televisão. Cada série de televisão tem os seguintes dados associados: o título; o criador; o número de temporadas; o número de episódios por temporada; a classificação etária; o ano de lançamento; o género; o elenco. Considere que cada série é univocamente identificado pelo título.

Para simplificar, considera-se que somente um cliente pode ter uma sessão aberta no sistema em cada instante. Para se executar um comando acessível a um cliente diferente do actual, é necessário que terminar a sessão actual e iniciar uma nova sessão.

2 Comandos

Nesta secção apresentam-se os vários comandos que o sistema deve ser capaz de interpretar e executar. Nos exemplos apresentados, diferenciamos o **texto escrito pelo utilizador** da retroacção escrita pelo programa na consola, ao executar o comando. Pode assumir que o utilizador não cometerá erros na introdução de argumentos nos comandos, para além dos descritos neste enunciado, ou seja, apenas tem de tratar as situações de erro descritas aqui, pela ordem que são descritas.

Na leitura de comandos o interpretador não deve fazer distinção entre maiúsculas e minúsculas. Por exemplo, para o comando **exit**, o interpretador deve aceitar como válidas as alternativas **EXIT**, **exit**, **Exit**, **ExIT**, etc. Nos vários exemplos que se seguem, o símbolo ↵ denota a mudança de linha.

Caso o utilizador introduza um comando inexistente, o programa escreve na consola a mensagem **Unknown command..** Por exemplo o comando inexistente **unknown** teria este efeito:

```
unknown↵
Unknown command.↵
```

Vários comandos têm argumentos. Pode assumir que o utilizador apenas usa argumentos de tipos correctos. No entanto, pode acontecer que os argumentos passados a um desses comandos tenham algum valor incorrecto. Por esse motivo, teremos de testar esses argumentos exactamente pela ordem especificada neste enunciado. Os argumentos são denotados com **este estilo**, na sua descrição.

2.1 Comando **upload**

Note que este comando será sempre o primeiro comando a ser executado não voltando a ser invocado durante a execução da aplicação.

Carrega base de dados de filmes e séries de televisão. Para tal, indica um número inteiro ≥ 1 de filmes a adicionar (**nMovie**), seguido da informação associada a cada um dos filmes. Para cada um dos filmes é indicado o **título**, o **nome do director**, a **duração em minutos**, a **classificação etária**, o **ano de lançamento**, o **género**, o número inteiro ≥ 1 de membros do elenco do filme (**nCast**) e os **nomes de cada um dos membros do elenco**. O comando indica de seguida um número inteiro ≥ 1 de séries de televisão a adicionar (**nSeries**), seguido da informação associada a cada uma das séries de televisão. Para cada uma das séries é indicado o **título**, o **nome do criador da série**, o **número de temporadas**, o **número de episódios por temporada**, a **classificação etária**, o **ano de lançamento**, o **género**, o número inteiro ≥ 1 de membros do elenco do filme (**nCast**) (**nCast**) e os **nomes de cada um dos membros do elenco**.

Pode assumir que neste comando não ocorrem erros e que todas as entradas estão bem-formadas. Como resultado deste comando são listados todos programas por ordem alfabética de título. Para cada filme, apresenta o título, o director, a duração, a classificação etária, o género e o elenco. Para cada série de televisão, apresenta o título, o criador, o número de temporadas, o número de episódios por temporada, a classificação etária, o género e o elenco. Ao listar o elenco de um filme ou série devem ser apresentados no máximo 3 membros do elenco, de acordo com a ordem com que foram lidos.

```

upload↵
1↵
Zootopia↵
Byron Howard↵
108↵
7+↵
2016↵
Animation↵
4↵
Ginnifer Goodwin↵
Jason Bateman↵
Idris Elba↵
Jenny Slate↵
2↵
Peaky Blinders↵
Steven Knight↵
5↵
6↵
16+↵
2013↵
Drama↵
7↵
Cillian Murphy↵
Paul Anderson↵
Helen McCrory↵
Sam Neill↵
Annabelle Wallis↵
Joe Cole↵
Tom Hardy↵
IT Crowd↵
Graham Linehan↵
4↵
6↵
16+↵
2006↵
Comedy↵
4↵
Chris O'Dowd↵
Richard Ayoade↵
Katherine Parkinson↵
Matt Berry↵
↵
Database was updated:↵
IT Crowd; Graham Linehan; 4; 6; 16+; 2016; Comedy; Chris O'Dowd; Richard Ayoade; Katherine Parkinson.↵
Peaky Blinders; Steven Knight; 5; 6; 16+; 2013; Drama; Cillian Murphy; Paul Anderson; Helen McCrory.↵
Zootopia; Byron Howard; 108; 7+; 2016; Animation; Ginnifer Goodwin; Jason Bateman; Idris Elba.↵

```

2.2 Comando **exit**

Termina a execução do programa. O comando não necessita de argumentos.

```

exit↵
Exiting...↵

```

Este comando tem sempre sucesso. Quando é invocado, o programa termina a sua execução.

2.3 Comando **register**

Regista uma nova conta. O comando inclui como parâmetros o **nome**, o **email**, a **senha de acesso** e o **dispositivo** que está a ser usado para aceder ao serviço. Em caso de sucesso é criada uma conta com o plano de adesão *Basic* e é iniciada uma nova sessão. O dispositivo usado no registo passa a estar associado à conta.

```
register↵
Alice↵
alice@gmail.com↵
pass1234↵
dev1↵
Welcome Alice (dev1).↵
```

Podem ocorrer os seguintes erros com o comando **register**:

- (1) Já existir um cliente com uma sessão aberta. A mensagem de erro apropriada será, nesse caso, (Another client is logged in.).
- (2) Já existir uma conta com o email indicado. A mensagem de erro apropriada será, nesse caso, (There is another account with email <email>.).

```
register↵
Alice↵
alice@gmail.com↵
pass1234↵
dev1↵
Welcome Alice (dev1).↵
register↵
Bob↵
bob@gmail.com↵
bob0000↵
dev2↵
Another client is logged in.↵
logout↵
Goodbye Alice (dev1 is still connected).↵
register↵
Mary↵
alice@gmail.com↵
pass1234↵
dev1↵
There is another account with email alice@gmail.com.↵
```

2.4 Comando **login**

Inicia a sessão de um cliente. O comando inclui os parâmetros **email**, a **senha** e o **dispositivo** usado para aceder ao serviço. Em caso de sucesso é iniciada a sessão associada à conta do utilizador. Caso o **dispositivo** esteja a ser usado pela primeira vez para aceder à conta, este será inserido no conjunto de dispositivos associados à conta. Caso contrário, o conjunto de dispositivos da conta não é alterado.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
```

Podem ocorrer os seguintes erros com o comando `login`:

- (1) O cliente já tiver uma sessão aberta. A mensagem de erro apropriada será, nesse caso, (Client already logged in.).
- (2) Já existir outro cliente com uma sessão aberta. A mensagem de erro apropriada será, nesse caso, (Another client is logged in.).
- (3) Não existir uma conta com o email indicado. A mensagem de erro apropriada será, nesse caso, (Account does not exist.).
- (4) A senha não corresponder à senha da conta. A mensagem de erro apropriada será, nesse caso, (Wrong password.).
- (5) Ter sido excedido o número de dispositivos da conta. A mensagem de erro apropriada será, nesse caso, (Not possible to connect more devices.).

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
login↵
alice@gmail.com↵
alice1234↵
dev2↵
Client already logged in.↵
login↵
bob@gmail.com↵
bob0000↵
dev2↵
Another client is logged in.↵
logout↵
Goodbye Alice (dev1 still connected).↵
login↵
mary@gmail.com↵
mmaarryyy↵
dev1↵
Account does not exist.↵
login↵
alice@gmail.com↵
1234alice↵
dev1↵
Wrong password.↵
```

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
logout↵
Goodbye Alice (dev1 still connected).↵
login↵
alice@gmail.com↵
alice1234↵
dev2↵
Not possible to connect more devices.↵
```

2.5 Comando **disconnect**

Termina a sessão de um cliente eliminando o acesso do dispositivo ao serviço. Esta operação não requer argumentos. Em caso de sucesso é dada como terminada a sessão corrente e o dispositivo deixa de pertencer ao conjunto de dispositivos associados à conta. Note que o acesso à conta com um dispositivo que foi previamente eliminado é similar a um acesso com um novo dispositivo.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
disconnect↵
Goodbye Alice (dev1 was disconnected).↵
```

Pode ocorrer o seguinte erro com o comando `disconnect`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
logout↵
Goodbye Alice (dev1 still connected).↵
disconnect↵
No client is logged in.↵
```

2.6 Comando **logout**

Termina a sessão de um cliente mantendo o acesso do dispositivo ao serviço. Esta operação não requer argumentos. Em caso de sucesso é dada como terminada a sessão actual.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
logout↵
Goodbye Alice (dev1 still connected).↵
```

Pode ocorrer o seguinte erro com o comando `logout`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
disconnect↵
Goodbye Alice (dev1 was disconnected).↵
logout↵
No client is logged in.↵
```

2.7 Comando `membership`

Altera o plano de adesão. O comando recebe como parâmetro o **nome do plano de adesão**. Pode assumir que o parâmetro é sempre um nome de plano válido. Em caso de sucesso é alterado o plano de adesão.

```
register↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
membership↵
Premium↵
Membership plan was changed from Basic to Premium.↵
```

Podem ocorrer os seguintes erros com o comando `membership`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) O novo plano de adesão ser igual ao plano actual. A mensagem de erro apropriada será, nesse caso, (No membership plan change.).
- (3) O novo plano de adesão não permitir o número actual de dispositivos conectados ou de perfis. A mensagem de erro apropriada será, nesse caso, (Cannot downgrade membership plan at the moment.).


```

membership↵
Premium↵
No client is logged in.↵
register↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
membership↵
Basic↵
No membership plan change.↵
membership↵
Premium↵
Membership plan was changed from Basic to Premium.↵
logout↵
Goodbye Alice (dev1 still connected).↵
login↵
alice@gmail.com↵
alice1234↵
dev2↵
Welcome Alice (dev2).↵
membership↵
Basic↵
Cannot downgrade membership plan at the moment.↵

```

2.8 Comando **profile**

Adiciona um perfil à conta. Esta operação recebe como argumentos o **nome do perfil** a adicionar, seguido do tipo de perfil (**NORMAL** ou **CHILDREN**). Caso o tipo de perfil seja *Criança* há o argumento adicional, um inteiro ≥ 5 , que indica a **classificação etária** do perfil. Em caso de sucesso é adicionado um novo perfil à conta da sessão actual.

```

login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
profile↵
my profile↵
NORMAL↵
New profile added.↵
profile↵
eve↵
CHILDREN↵
5↵
New profile added.↵

```

Podem ocorrer os seguintes erros com o comando **profile**:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).

- (2) Já existir um perfil com o nome dado. A mensagem de erro apropriada será, nesse caso, (There is already a profile <name>.).
- (3) Ter excedido o número de perfis permitidos pelo plano de adesão. A mensagem de erro apropriada será, nesse caso, (Not possible to add more profiles.).

```

profile↵
my profile↵
NORMAL↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
profile↵
my profile↵
NORMAL↵
New profile added.↵
profile↵
my profile↵
CHILDREN↵
10↵
There is already a profile my profile.↵
profile↵
eve↵
CHILDREN↵
5↵
New profile added.↵
profile↵
bob↵
NORMAL↵
Not possible to add more profiles.↵

```

2.9 Comando **select**

Seleciona um perfil da conta. Esta operação recebe como argumento o **nome do perfil** a aceder. Em caso de sucesso é o cliente é direcionado para o perfil selecionado.

```

login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
profile↵
bob↵
NORMAL↵
New profile added.↵
select↵
bob↵
Welcome bob.↵

```

Podem ocorrer os seguintes erros com o comando **select**:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) Não existir na conta o perfil indicado. A mensagem de erro apropriada será, nesse caso, (Profile does not exist.).

```
select
my profile↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
profile↵
bob↵
NORMAL↵
New profile added.↵
select↵
eve↵
Profile does not exist.↵
```

2.10 Comando **watch**

Visualização de um filme ou série de televisão. Esta operação recebe como argumento o **título** do filme ou série de televisão a ver. Em caso de sucesso é registada a visualização do programa na lista dos programas vistos recentemente.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
select↵
bob↵
Welcome bob.↵
watch↵
Zootopia↵
Loading Zootopia...↵
```

Podem ocorrer os seguintes erros com o comando **watch**:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) Não existir um perfil seleccionado. A mensagem de erro apropriada será, nesse caso, (No profile is selected.).
- (3) Não existir um filme ou série de televisão com o título dado. A mensagem de erro apropriada será, nesse caso, (Show does not exist.).
- (4) O filme ou série de televisão não for adequado para a classificação etária do perfil. A mensagem de erro apropriada será, nesse caso, (Show not available.).

```

watch↵
Zootopia↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
watch↵
IT Crowd↵
No profile is selected.↵
profile↵
eve↵
CHILDREN↵
5↵
New profile added.↵
select↵
eve↵
watch↵
Moana↵
Show does not exist.↵
watch↵
IT Crowd↵
Show not available.↵

```

2.11 Comando **rate**

Classifica um filme ou série de televisão visto recentemente. Esta operação recebe como argumentos o **título** do filme ou série de televisão a classificar e um inteiro entre 1 e 5 que indica a **classificação** atribuída. Em caso de sucesso o programa a classificação é registada.

```

login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
select↵
eve↵
Welcome eve.↵
watch↵
Zootopia↵
Loading Zootopia...↵
rate↵
Zootopia↵
5↵
Thank you for rating Zootopia.↵

```

Podem ocorrer os seguintes erros com o comando **rate**:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) Não existir um perfil seleccionado. A mensagem de erro apropriada será, nesse caso, (No profile is selected.).

- (3) Não existir um filme ou série de televisão com o título dado. A mensagem de erro apropriada será, nesse caso, (Show does not exist.).
- (4) O programa não estar na lista dos programas vistos recentemente. A mensagem de erro apropriada será, nesse caso, (Can only rate recently seen shows.).
- (5) O programa já ter sido classificado. A mensagem de erro apropriada será, nesse caso, (Show already rated.).

```

rate↵
Zootopia↵
5↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
rate↵
IT Crowd↵
4↵
No profile is selected.↵
select↵
bob↵
Welcome bob.↵
rate↵
Moana↵
4↵
Show does not exist.↵
rate↵
IT Crowd↵
4↵
Can only rate recently seen shows.↵
watch↵
IT Crowd↵
Loading IT Crowd...↵
rate↵
IT Crowd↵
4↵
Thank you for rating IT Crowd.↵
rate↵
IT Crowd↵
4↵
Show already rated.↵

```

2.12 Comando **infoAccount**

Mostra toda a informação associada a uma conta. Esta operação não requer argumentos. Em caso de sucesso é apresentada a seguinte informação: nome da conta, plano de adesão, os dispositivos conectados (listar por ordem de antiguidade), seguido da informação detalhada sobre cada perfil pela ordem com que os perfis foram criados. Caso não existam perfis definidos é mostrada a mensagem (No profiles defined.). Para cada perfil deve ser mostrado o nome do perfil, a classificação etária caso seja um perfil *Children*, a listagem dos programas vistos

recentemente (listar por ordem de antiguidade) e a listagem das classificações atribuídas (listar por ordem de inserção). Caso não a lista dos programa recentes seja vazia deve ser mostrada a mensagem (Empty list of recently seen shows.). Note que neste caso também não existem classificações atribuídas. Caso não existam classificações atribuídas não é mostrada qualquer mensagem.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
infoAccount↵
Alice:↵
Standard (dev1; dev2).↵
Profile: alice↵
Empty list of recently seen shows.↵
Profile: eve (12)↵
Star Wars: The Last Jedi; Paddington 2.↵
...
infoAccount↵
Alice:↵
Standard (dev1; dev2).↵
Profile: alice↵
Star Wars: The Last Jedi; IT Crowd; The Crown; Doctor Who; Sherlock.↵
Sherlock (4); Star Wars: The Last Jedi (5); Doctor Who (3).↵
Profile: eve (12)↵
Zootopia; Star Wars: The Last Jedi; Paddington 2.↵
Paddington 2 (5); Zootopia (5); Star Wars: The Last Jedi (3).↵
disconnect↵
Goodbye Alice (dev1 was disconnected).↵
login↵
bob@gmail.com↵
bob0000↵
dev5↵
Welcome Bob (dev5).↵
infoAccount↵
Bob:↵
Basic (dev5).↵
No profiles defined.↵
```

Pode ocorrer o seguinte erro com o comando `infoAccount`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).

```
infoAccount↵
No client is logged in.↵
```

2.13 Comando `searchByGenre`

Lista os filmes e séries de televisão de um dado género. Esta operação recebe como argumento o **género** dos filmes e programas de televisão a listar. Os programas devem ser

listadas por ordem alfabética do título. Neste comando devem ser mostrados todos os programas que satisfazem o critério de procura independentemente da classificação etária do perfil.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
select↵
alice↵
searchByGenre↵
Science Fiction↵
Elysion; Neill Blomkamp; 109; 16+; 2013; Science Fiction, Matt Damon; Jodie Foster; Sharlto Copley;
Alice Braga; Diego Luna.↵
Star Wars: The Last Jedi; Rian Johnson; 152; 7+; 2016; Science Fiction, Mark Hamill; Carrie Fisher;
Daisy Ridley; John Boyega.↵
Stranger Things; The Duffer Brothers; 2; 8; 16+; 2016; Science Fiction, Winona Ryder; David Harbour;
Millie Bobby Brown.↵
```

Podem ocorrer os seguintes erros com o comando `searchByGenre`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) Não existir um perfil seleccionado. A mensagem de erro apropriada será, nesse caso, (No profile is selected.).
- (3) Não existir nenhum programa do género indicado. A mensagem de erro apropriada será, nesse caso, (No show found.).

```
searchByGenre↵
Science fiction↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
searchByGenre↵
Science fiction↵
No profile is selected.↵
select↵
bob↵
Welcome bob.↵
searchByGenre↵
Psycho-Biddy↵
No show found.↵
```

2.14 Comando `searchByName`

Lista os filmes e séries de televisão em que tenha participado o membro do elenco, realizador ou criador com o nome dado. A listagem deve estar ordenada por ordem cronológica dos programas e, em caso de empate, deve ser usada a ordem alfabética do título.

Neste comando devem ser mostrados todos os programas que satisfazem o critério de procura independentemente da classificação etária do perfil.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
select↵
alice↵
searchByName↵
Benedict Cumberbatch↵
Doctor Strange; Scott Derrickson; 115; 12+; 2016; Fantasy; Benedict Cumberbatch; Chiwetel Ejiofor;
Rachel McAdams; Benedict Wong.↵
Sherlock; Mark Gatiss; 4; 4; 16+; 2010; Crime; Benedict Cumberbatch; Martin Freeman; Una Stubbs;
Louise Brealey.↵
```

Podem ocorrer os seguintes erros com o comando `searchByName`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) Não existir um perfil seleccionado. A mensagem de erro apropriada será, nesse caso, (No profile is selected.).
- (3) Não existir nenhum programa com o participante indicado. A mensagem de erro apropriada será, nesse caso, (No show found.).

```
searchByName↵
Benedict Cumberbatch↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
searchByName↵
Benedict Cumberbatch↵
No profile is selected.↵
select↵
bob↵
Welcome bob.↵
searchByName↵
Jar Jar Binks↵
No show found.↵
```

2.15 Comando `searchByRate`

Lista os filmes e séries de televisão com a classificação igual a superior ao valor dado. A listagem apresenta os filmes ordenados por ordem decrescente de classificação. Filmes com a mesma classificação devem ser ordenados por ordem alfabética crescente do título. A classificação de um filme é a média das classificações atribuídas devendo ser apresentado o valor com uma casa decimal sem arredondamento. Por exemplo, se um filme recebeu as classificações

5, 3, 3, 4, deve ser apresentada a classificação **3.7**. Neste comando devem apenas ser mostrados programas adequados à classificação etária do perfil.

```
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
select↵
alice↵
searchByRate↵
4↵
Stranger Things; The Duffer Brothers; 2; 8; 16+; Horror, Winona Ryder; David Harbour; Millie Bobby Brown. [5.0]↵
Sherlock; Mark Gatiss; 4; 4; 16+; 2010; Crime; Benedict Cumberbatch; Martin Freeman; Una Stubbs;
Louise Brealey. [4.7]↵
Zootopia; Byron Howard; 108; 7+; 2016; Animation; Ginnifer Goodwin; Jason Bateman; Idris Elba; Jenny Slate. [4.0]↵
```

Podem ocorrer os seguintes erros com o comando `searchByRate`:

- (1) Não existir uma sessão inicializada. A mensagem de erro apropriada será, nesse caso, (No client is logged in.).
- (2) Não existir um perfil seleccionado. A mensagem de erro apropriada será, nesse caso, (No profile is selected.).
- (3) Não existir nenhum programa com classificação igual ou superior ao valor indicado. A mensagem de erro apropriada será, nesse caso, (No show found.).

```
searchByRate↵
4↵
No client is logged in.↵
login↵
alice@gmail.com↵
alice1234↵
dev1↵
Welcome Alice (dev1).↵
searchByRate↵
4↵
No profile is selected.↵
select↵
bob↵
Welcome bob.↵
searchByRate↵
5↵
No show found.↵
```

3 Desenvolvimento do Projeto

A sua aplicação deve tirar o melhor partido possível da matéria leccionada. Em particular, fazê-la **o mais extensível possível**. Deve ser construída de modo a **minimizar as alterações necessárias**, caso se pretendam acrescentar, mais tarde, **novos tipos de participantes**, ou de **recursos associados a gravações**).

Comece por desenvolver a interface principal da sua aplicação, identificando claramente quais os comandos que a sua aplicação deve suportar, bem como quais as entradas e saídas, não esquecendo as pré-condições. Depois, identifique as entidades de que vai necessitar para implementar este sistema. Identifique e especifique cuidadosamente as **interfaces** e **classes** de que necessita. Deve documentar a sua concepção e o seu desenvolvimento quer através de um diagrama de classes e interfaces, quer através de documentação adequada no seu código.

Construa o esqueleto da classe `Main` que trata da entrada e saída dos dados e da interacção com a aplicação. É normal que no princípio o seu programa ainda não faça tudo. Lembre-se da **regra da versão estável**: não tente fazer tudo de uma só vez. Vá fazendo, testando, e avançando por incrementos, à medida que as funcionalidades vão sendo desenvolvidas. Se necessário, crie pequenos programas de teste auxiliares. Desenvolva também as operações de forma incremental.

4 Submissão ao Mooshak

Para submeter o seu programa ao Mooshak registe-se no concurso POO1718-TP2 e siga as instruções apresentadas na área Moodle da disciplina.

4.1 Sintaxe dos comandos

Note que para cada comando será apresentada apenas uma única mensagem de saída. Ou seja, as condições que causam falhas num comando devem ser verificadas pela ordem descrita no enunciado e assim que uma dessas condições se verificar não é necessário verificar as condições seguintes. Por exemplo, se ao executar o comando `login` a senha da conta estiver errada, deve ignorar o resto das condições de validação, devolvendo a mensagem de erro apropriada ao primeiro erro detectado, seguindo a ordem pela qual os vários tipos de erros de input estão apresentados, nesta especificação.

4.2 Testes

Os testes do Mooshak verificam de forma incremental a implementação dos vários comandos e serão publicados a 13 de Maio.

Ficheiro de teste `01_in.txt` (10 pontos)

Comandos testados: `upload` e `exit`

Testa os comandos `upload` e `exit`.

Ficheiro de teste `02_in.txt` (10 pontos)

Comandos testados: `register`, `logout`, `disconnect`, `exit`

Testa os comandos sem situações de erro.

Ficheiro de teste `03_in.txt` (5 pontos)

Comandos testados: `register`, `logout`, `disconnect`, `exit`

Testa os comandos com situações de erro.

Ficheiro de teste `04_in.txt` (5 pontos)

Comandos testados: `login`, `register`, `logout`, `disconnect`, `exit`

Testa o comando sem situações de erro.

Ficheiro de teste `05_in.txt` (5 pontos)

Comandos testados: `login`, `register`, `logout`, `disconnect`, `exit`

Testa o comando com situações de erro.

Ficheiro de teste 06_in.txt (10 pontos)

Comandos testados: **membership**, **profile**, **select**, etc.

Testa os comandos sem situações de erro.

Ficheiro de teste 07_in.txt (5 pontos)

Comandos testados: **membership**, **profile**, **select**, etc.

Testa os comandos com situações de erro.

Ficheiro de teste 08_in.txt (15 pontos)

Comandos testados: **watch**, **rate**, **infoAccount**, etc.

Testa os comandos sem situações de erro.

Ficheiro de teste 09_in.txt (5 pontos)

Comandos testados: **watch**, **rate**, **infoAccount**, etc.

Testa os comandos com situações de erro.

Ficheiro de teste 10_in.txt (10 pontos)

Comandos testados: **searchByGenre**, **searchByName**, etc.

Testa os comandos sem situações de erro.

Ficheiro de teste 11_in.txt (5 pontos)

Comandos testados: **searchByGenre**, **searchByName**, etc.

Testa os comandos com situações de erro.

Ficheiro de teste 12_in.txt (10 pontos)

Comandos testados: **searchByRate**, etc.

Testa os comandos sem situações de erro.

Ficheiro de teste 13_in.txt (5 pontos)

Comandos testados: **searchByRate**, etc.

Testa os comandos com situações de erro.