

Exercícios POO

Universidade NOVA de Lisboa, Departamento de Informática - Equipa de POO

4 Desenvolvimento da aplicação CloudSharing

4.1 Descrição do problema

O objectivo deste trabalho é o desenvolvimento de uma aplicação que permita a gestão de um sistema para armazenar ficheiros na cloud (e.g. Dropbox). Cada utilizador registado tem acesso a uma conta onde pode adicionar ficheiros, sendo que esta conta tem uma **capacidade máxima** que não pode ser excedida. Os ficheiros armazenados no sistema podem ser **consultados através de um website**. Para além disso, também é possível **partilhar ficheiros com outros utilizadores** registados.

Cada **conta** (ou utilizador) tem um **email** (identificador único da conta) e os **ficheiros dos quais é proprietário**. Existem dois tipos de contas, as contas básicas gratuitas (“**basic**”) e as contas premium com anuidade (“**premium**”). As contas **premium têm 5GB de espaço** e podem **partilhar os seus ficheiros com outros utilizadores registados**. As contas **básicas têm 2GB de espaço** e os seus ficheiros **não podem ser partilhados**. **Considere que 1GB corresponde a 1024MB**. Outra diferença entre as contas básicas e premium, é a contabilização do espaço usado. **Nas contas premium o espaço ocupado pelos ficheiros partilhados não é contabilizado, enquanto nas contas básicas o espaço ocupado pelos ficheiros partilhados é contabilizado a 50%**. Por exemplo, para uma conta básica com 100MB em ficheiros proprietários e 80MB em ficheiros partilhados, considera-se que esta conta usa 140MB dos 2GB disponíveis.

Cada **ficheiro** tem um **nome**, uma **dimensão** (em MB) e uma **conta** (ou utilizador) **proprietário**. Um ficheiro é **identificado univocamente pelo seu nome e pela conta a que pertence**.

A aplicação deve então permitir:

1. Criar uma conta (comando **ADD**). São fornecidos o nome da conta (o email do utilizador) e o tipo de conta (básica ou premium). Em caso de sucesso a conta é adicionada (“Account was added.”). A operação falha se: (1) já existir uma conta com esse email (“Account already exists.”).
2. Adicionar um ficheiro (comando **UPLOAD**). São fornecidos o nome da conta, o nome do ficheiro e a dimensão do ficheiro em MB. Em caso de sucesso o ficheiro é adicionado à conta (“File uploaded into account.”). A operação falha se: (1) a conta não existir (“Account does not exist.”); (2) já existir um ficheiro com esse nome nessa conta (“File already exists in the account.”); ou (3) a dimensão do ficheiro exceder a capacidade da conta (“File size exceeds account capacity.”).
3. Partilhar um ficheiro (comando **SHARE**). São fornecidos o nome da conta a que o ficheiro pertence, o nome da conta a adicionar à partilha e o nome do ficheiro. Em caso de sucesso o ficheiro é partilhado (“File was shared.”). A operação falha se: (1) uma das contas não existir (“Account does not exist.”); (2) o ficheiro não existir (“File does not exist.”); (3) a conta não permitir partilhar ficheiros (“Account does not allow file sharing.”); (4) a partilha já existir (“File already shared.”); ou (5) a dimensão do ficheiro exceder a capacidade da conta (“File size exceeds account capacity.”).

4. Listar todos os ficheiros proprietários e partilhados de uma conta (comando `LISTFILES`). É fornecido o nome da conta. Em caso de sucesso a operação apresenta o cabeçalho (“Account files:”) seguido do nome de cada ficheiro, a sua dimensão e se é partilhado. A operação falha se: (1) a conta não existir (“Account does not exist.”).
5. Listar todas as contas (comando `LISTALL`). A operação tem sempre sucesso e apresenta o cabeçalho (“All accounts:”) seguido do nome de cada conta e o tipo de conta.
6. Sair da aplicação (comando `EXIT`). A operação tem sempre sucesso (“Exiting...”).

4.2 Exemplo de interacção com a aplicação

Desenvolva a sua aplicação para que esta garanta, pelo menos, o modelo de interacção ilustrado no exemplo seguinte (o carácter `↵` representa uma mudança de linha):

```
ADD cf@gmail.com premium↵
Account was added.↵
↵
ADD cf@gmail.com basic↵
Account already exists.↵
↵
UPLOAD cf@gmail.com poo.txt 2↵
File uploaded into account.↵
↵
UPLOAD poo@gmail.com poo.txt 2↵
Account does not exist.↵
↵
UPLOAD cf@gmail.com poo.txt 2↵
File already exists in the account.↵
↵
UPLOAD cf@gmail.com ada.pdf 7000↵
File size exceeds account capacity.↵
↵
UPLOAD cf@gmail.com contactBook.pdf 3000↵
File uploaded into account.↵
↵
ADD mg@fct.unl.pt basic↵
Account was added.↵
↵
ADD am@fct.unl.pt premium↵
Account was added.↵
↵
SHARE cf@gmail.com am@fct.unl.pt poo.txt↵
File was shared.↵
UPLOAD am@fct.unl.pt stupidFriendsBook.pdf 2000↵
File uploaded into account.↵
↵
```

```

SHARE am@fct.unl.pt bf@fct.unl.pt zoo.zip↵
Account does not exist.↵
↵
UPLOAD mg@fct.unl.pt zoo.zip 1000↵
File uploaded into account.↵
↵
SHARE mg@fct.unl.pt am@fct.unl.pt zoo.zip↵
Account does not allow file sharing.↵
↵
SHARE cf@gmail.com mg@fct.unl.pt contactBook.pdf↵
File size exceeds account capacity.↵
↵
SHARE cf@gmail.com am@fct.unl.pt poo.txt↵
File already exists in the account.↵ como é que o comando SHARE pode devolver esta string se não está nas opções??
↵
LISTFILES am@fct.unl.pt↵
Account files:↵
poo.txt (2 MB) (shared)↵
stupidFriendsBook.pdf (2000 MB)↵
↵
LISTALL ↵
All accounts:↵
cf@gmail.com (Premium)↵
mg@fct.unl.pt (Basic)↵
am@fct.unl.pt (Premium)↵
↵
EXIT↵
Exiting...↵
↵

```

4.3 Desenvolvimento

Desenvolva a sua aplicação de acordo com as seguintes fases:

1. Desenvolva o(s) interface(s) de suporte à aplicação. Utilize o conceito de polimorfismo. Desenhe um diagrama de classes e interfaces para ilustrar a proposta de modelação.
2. Implemente e teste a aplicação.
3. Submeta o código fonte da aplicação ao *Mooshak*.

Ficheiros de testes

Os testes do Mooshak verificam de forma incremental a implementação dos vários comandos:

- Ficheiro de teste: `1_in_base_add.txt` (20 pontos)
Comandos testados: `ADD`, `LISTALL`, `EXIT`
Contexto: não são testadas as condições onde os comandos podem falhar.
- Ficheiro de teste: `2_in_base_upload.txt` (20 pontos)
Comandos testados: todos os comandos do teste 1 e os comandos `UPLOAD`, `LISTFILES`
Contexto: não são testadas as condições onde os comandos podem falhar.
- Ficheiro de teste: `3_in_base_share.txt` (20 pontos)
Comandos testados: todos os comandos do teste 2 e o comando `SHARE`
Contexto: não são testadas as condições onde o comando pode falhar.
não está a ser usado a regra de ocupar 50% do tamanho original???

- Ficheiro de teste: `4_in_pre_add.txt` (10 pontos)
Comandos testados: `ADD`
Contexto: São testadas todas as condições em que o comando `ADD` pode falhar.
- Ficheiros de teste: `5_in_pre_upload.txt` (10 pontos)
Comandos testados: `UPLOAD`
Contexto: São testadas todas as condições em que o comando `UPLOAD` pode falhar.
- Ficheiros de teste: `6_in_pre_share.txt` (10 pontos)
Comandos testados: `SHARE`
Contexto: São testadas todas as condições em que o comando `SHARE` pode falhar.
- Ficheiro de teste: `7_in_pre_files.txt` (10 pontos)
Comandos testados: `LISTFILE`
Contexto: São testadas todas as condições em que o comando `LISTFILES` pode falhar.