

UPskill – Java+.NET

Programação Orientada a Objetos – Ficha Prática 12

EXERCÍCIO PLATAFORMA DE STREAMING

1. A ficha de trabalho consiste em desenvolver uma API para suportar um website de streaming de filmes que permita aos utilizadores aceder a uma biblioteca de filmes e séries, organizados por categorias. O objetivo é implementar um programa em C# que incorpore os vários conceitos abordados até então (Classes, Herança, Polimorfismo, Composição/Agregação, Coleções, Interfaces, Tipos Enumerados e Exceções). Considere que:
 - a. Cada conteúdo possui um Título, Ano de lançamento, Classificação Etária (Livre, 12+, 16+, 18+) e um resumo.
 - b. Os conteúdos podem ser Filmes ou Séries. Os filmes possuem uma duração em minutos enquanto as séries possuem o número de temporadas e número de episódios.
 - c. Cada conteúdo deve pertencer a uma ou mais categorias, permitindo uma estrutura flexível para facilitar a navegação no catálogo. Devem existir categorias padrão (como ação, terror ou comédia) e categorias específicas para organizar os conteúdos (destaques, clássicos, Halloween, etc). O conteúdo deverá:
 - i. Permitir a adição de uma categoria, devendo despoletar uma exceção caso a categoria já exista.
 - ii. Remover uma categoria, devendo despoletar uma exceção caso a categoria não exista.
 - iii. Devolver um array de categorias
 - d. Cada utilizador da plataforma é caracterizado pelo seu nome de utilizador, email e país. Deve ainda ser possível:
 - i. Gerir a subscrição do utilizador que pode ser trial (refere-se à primeira subscrição válida por 15 dias), ativo, inativo. Deve ser possível registar o histórico de subscrições, devendo ser registado cada mês que o utilizador subscreveu a plataforma. Cada subscrição tem uma duração de 30 dias a contar da data da ultima subscrição. Se for adicionada uma subscrição quando ainda existe uma subscrição ativa deverá ser lançada uma SubscricaoAindaAtivaException (e não ser que essa adição seja realizada nos últimos 3 dias da subscrição).
 - ii. O utilizador tem uma lista de favoritos, podendo:
 1. Adicionar um conteúdo à lista de favoritos. Se o conteúdo já estiver na lista de favoritos deverá ser despoletada uma exceção.

2. Remover um conteúdo da lista de favoritos. Se o conteúdo não existir na lista deverá ser despoletada uma exceção.
 3. Devolver um array de conteúdos da lista de favoritos.
 4. Os conteúdos com mais de 6 meses na lista de favoritos devem ser removidos automaticamente. Para isso, deverá ter em consideração que terá de associar uma data de registo do conteúdo nos favoritos.
- iii. Deverá ser possível registar as visualizações do utilizador (conteúdo e data). Só deve ser possível registar a visualização se a subscrição estiver ativa (despoletando uma exceção se não estiver ativa). Para os filmes deverá ser registado o tempo de visualização e para as séries, para além do tempo de visualização é necessário registar a temporada, episódio e minuto em que o utilizador parou de ver. É apenas registado o último registo de histórico para um determinado conteúdo.
2. Recorrendo a teste unitários, teste de modo abrangente o código desenvolvido.
 3. De modo a verificar a sua implementação e os conhecimentos adquiridos, analise o código disponibilizado. Preste atenção na estrutura, nas declarações de classe e nos métodos.

Notas para os estudantes: Reserve um tempo para ler e entender cuidadosamente cada parte do código. Sinta-se à vontade para experimentar modificações ou funcionalidades adicionais para aprofundar a sua compreensão. Esta ficha de trabalho foi concebida para encorajar os alunos a trabalhar ativamente com o código fornecido, identificar conceitos-chave e aplicar os seus conhecimentos para alargar o projeto. O seu objetivo é reforçar a compreensão das classes abstratas, do polimorfismo e da herança num contexto prático.