

Projecto Web Services

Descrição

O sistema será uma base de dados de livros, em que os membros da comunidade poderão indicar os livros que tem, e os que gostavam de ter, e destes os que leram e qual a sua pontuação

Os utilizadores podem fornecer o seu id de utilizador a outros, e assim podem ser adicionados como amigos de outro utilizador. Os amigos de um utilizador podem consultar a wishlist deste utilizador.

Utilizadores

Dados de utilizador:

- id *
- Nome *
- Email *
- Palavra-passe *
- Perfil (administrador, leitor) *
- Data de nascimento

Gestão de Livros

O sistema terá uma base de dados de livros que é mantida por todos os utilizadores, todos podem adicionar livros, tendo estes que ser aprovados por um administrador antes de ficarem disponíveis para consulta por outros membros da comunidade. Depois de um livro ser adicionado e aprovado, apenas os administradores podem modificar o livro, mas terá que ficar registado quem inseriu o livro e que o aprovou. Não podem ser adicionados livros duplicados, e para tal é usado o ISBN do livro para garantir que já não existe em sistema.

Será necessário manter esta informação e como seria de esperar, os utilizadores podem gerir os seus livros, dizendo que já leram um livro que tinham, ou que já possuem um livro que queriam ter, e neste caso o livro deixa de estar na wishlist do utilizador.

Uma vez existente no sistema o livro, qualquer utilizador da comunidade pode fazer o seguinte:

- Indicar que possui este livro
- Indicar que gostava de ter este livro
- Indicar que já leu o livro
- E caso já tenha lido o livro poderá dar uma classificação

As classificações de filmes que tem que ficar registada a seguinte informação:

- Quem fez a classificação
- Qual a classificação de 1-10
- Quando foi feita a classificação
- Qual o livro que está a ser classificado

Informação obrigatória a manter do livro

- Nome
- Autor(es)
- Género(s)
- Descrição
- ISBN
- Quem registou o livro
- Quem aprovou o livro
- Capa / imagem
- Classificação (média de todas as classificações existentes arredondado a 1 casa decimal)

Como este sistema tem níveis de permissões, para cada método terá que ser fornecido um parâmetro que é o utilizador que invoca os métodos, por exemplo my_user_id.

API – Funções

Utilizadores

Método	Descrição
addUser	adicionar um utilizador com os campos de acordo com o indicado anteriormente <ul style="list-style-type: none">• Apenas disponível para utilizadores administradores, portanto o servidor rest terá que validar se o user_id fornecido é um administrador.• O primeiro utilizador a criado devem fazê-lo directamente na base de dados para poderem testar.
editUser	Modificar um utilizador existente. Isto implica mudar os campos nome, email, palavra-passe e perfil. <ul style="list-style-type: none">• A alteração de perfil apenas pode ser feita por um administrador.• Podem escolher por validar esta operação no controlador ou ter uma função para modificar o perfil de utilizador.• Esta função deverá modificar apenas os dados fornecidos pelo que o servidor terá que tratar esta situação de acordo.
changeUserStatus	Esta função altera o estado de um utilizador de activo para inactivo, ou vice-versa. Apenas administradores podem utilizar esta funcionalidade
getUser	Retorna a lista de todos os utilizadores ou se for fornecido um Id, os dados deste utilizador. Apenas os administradores podem ver a lista de utilizadores do sistema. Apenas administradores e o próprio utilizador pode ver os dados, não pode ver de mais ninguém. Os dados de utilizador devem ter indicação dos livros que possui, que leu, que estão na wishlist e que ele avaliou.
addFriend	Permite adicionar um utilizador à lista de amigos
getWishList	Permite obter a wishlist de um utilizador. Apenas o próprio e amigos do próprio é que podem ver a wishlist

Livros

Método	Descrição
getBooks	Devolve a lista de livros aprovados no sistema. Caso seja administrador devolve a lista de todos os filmes existentes.
getBookInfo	Devolve a informação referente a um livro cujo id será fornecido por parâmetro. Caso o utilizador seja administrador inclui livros não aprovados, se for um leitor, não pode mostrar dados de livros não aprovados a não ser que tenha sido um livro criado pelo próprio. Na informação do livro, deve vir mencionado a classificação, bem como o número de pessoas que tem o livro, e o número de pessoas que leram o livro, e o número de classificações que teve.
editBook	Esta função modifica os dados de um livro e apenas está disponível para administradores. Esta função permite aprovar um livro também alterando o seu estado. <ul style="list-style-type: none">• Esta função deverá modificar apenas os dados fornecidos pelo que o servidor terá que tratar esta situação de acordo.• O ISBN não pode ser alterado.
setOwned	Permite indicar se o utilizador possui um livro ou não.
setWished	Permite indicar se o utilizador gostava de ter este livro ou não.
setRead	Permite indicar se o utilizador já leu o livro ou não.
rateBook	Permite dar uma classificação a um livro. Apenas podem classificar livros utilizadores que tenham este livro na sua lista de livros lidos.
searchBook	Permite pesquisar um livro no sistema, e os critérios passíveis de serem pesquisados são nome, género, autor, ISBN. Deverá devolver os livros.

Cada função implementada no servidor deverá validar os tipos de dados dos parâmetros, bem como a sua obrigatoriedade ou não e dar mensagem de retorno adequada.

Quando ocorrer algum tipo de erro a função deverá retornar um id menor que zero e a descrição do erro. Se a função executar normalmente e for uma inserção em base de dados o id deverá ser o id do objecto inserido. Em caso de funções que não tenham inserção o retorno em caso de sucesso para o id deverá ser 0, se aplicável. Uma boa prática é todas as funções terem no retorno um id de erro ou sucesso, uma mensagem, e mais quaisquer outros campos relevantes.

Requisitos de implementação

O projecto é para ser realizado em CodeIgniter e deverá utilizar Base de Dados MySQL. No cliente terão que implementar formulários e views preferencialmente com Bootstrap local (não CDN) para cada processo a ser realizado pelo cliente.

Entregáveis

Devem enviar por email para jose.em.melo@uac.pt com assunto "Projecto Web Services" o seguinte até às 9h GMT-1 de dia 4 de Janeiro de 2019:

- Nome dos elementos do grupo
- Modelo de dados da Base de dados realizado em MySQL Workbench
- Dump da base de dados que utilizaram no projecto

- Especificação da API indicando para cada método acima mencionado
 - A. Parâmetros
 - nome
 - indicação se são obrigatórios
 - qual o tipo de dados
 - B. Método HTTP
 - C. Retorno do método
 - D. A especificação pode ser em PDF ou mesmo em HTML no próprio projecto servidor, para isso será necessário indicar o URL de acesso à mesma.
- ZIP contendo o projecto cliente
- ZIP contendo o projecto servidor