

Biblioteca - Desafio

Descrição de todo o projeto

Criar um sistema para uma biblioteca gerenciar a locação de seus livros

Link ViaCep: <https://viacep.com.br/ws/01001000/json>

| | |
|--|--|
| Back End | |
| Prover um webservices com arquitetura REST para atender as demandas do projeto | |
| Modelagem de Dados: | Nesta fase avaliamos o desempenho em organização do projeto, domínio do negócio, persistência de dados |
| Para iniciar será necessário o Cadastro contendo os campos, Id, Nome, CPF, Email, Telefone, Login, Senha | |
| O campo login pode conter caracteres que representam cpf, telefone ou apelido com até 20 caracteres | |
| Não poderá ser duplicado os campos cpf e login na base de dados | |
| Para o cadastro deverá ter o seu endereço através uma tabela de endereço com os campos: Id, Cep, Logradouro, Numero, Bairro, Localidade, Uf, IBGE. NOTA: O sistema deverá utilizar de algum client API Java como RestTemplate e FeingClient para buscar um endereço do serviço via cep conforme link: | |
| Será necessário também cadastrar os livros com os campos: Id, ISBN, Titulo, Valor da Diaria, Número de exemplares do livro e Número dos exemplares reservados. | |
| Para realizar a locação deverá armazenar as informações abaixo: Id, Data Agendamento, Data Retirada, Data Finalizacao, Valor Total e o Status Locação (RESERVADA, EFETIVADA, FINALIZADA) | |
| Como será possível retirar mais de um livro em cada Locação, deverá registrar estes livros selecionados incluindo os campos: Data Previsao Entrega, Data Entrega, Numero de diarias (data entrega - data retirada), Valor Diaria, Valor Locação (valor diaria * diarias) | |
| Regra de negócio: | |
| <div><div>1.</div><div>O campo login pode conter caracteres que representam cpf, telefone ou apelido com até 20 caracteres</div></div> <div><div>2.</div><div>Não poderá ser duplicado os campos cpf e login na base de dados</div></div> <div><div>3.</div><div>Todo cadastro precisará de um endereço</div></div> <div><div>4.</div><div>Toda locação iniciará com Data Agendamento, Status = RESERVADA e Valor Total = 0,0. Somente o campo Data Retirada aceitará nulo</div></div> <div><div>5.</div><div>Na inclusão dos itens de livro na locação, deverá ser considerado se o campo número de exemplares do livro maior ou igual a 1</div></div> <div><div>6.</div><div>A cada inclusão de livro na locação deverá ser informada uma previsão de data de entrega e manipular os campos do livro da seguinte maneira.<div><div>a.</div><div>Decrementar o campo exemplares em 1</div></div><div><div>b.</div><div>Incrementar o campo reservados em 1</div></div></div></div> <div><div>7.</div><div>Toda vez que for selecionada uma locação através da consulta de locações, deverá ser acionado a rotina que analisa todos os livros da locação atualizando os campos: número de diárias já realizadas, valor da locação de cada livro com as seguintes regras:</div></div> | |

| | |
|--|---|
| <ul style="list-style-type: none"> a. o cálculo só será executado em livros que ainda não entregues (Data Entrega igual null) b. Cálculo de número de diárias : Data de Hoje - Data Retirada (Locacao) c. Cálculo de valor de locação: Numero de Diárias * Valor Diária d. Valor Total da Locação é o resultante da soma do Valor Locação de todos os livros da locação. <p>8. Quando o usuário vier retirar os livros atualizar o campo Data de Retirada e mudar o Status para EFETIVADA</p> <p>9. Quando o usuário entregar todos os livros da locação, atualizar o campo Data Finalização e Status para FINALIZADA e atualizar os campos do livro conforme abaixo:</p> <ul style="list-style-type: none"> a. Incrementar o campo exemplares em 1 b. Decrementar o campo reservados em 1 | |
| Disponibilizar uma consulta de locações pelos parâmetros: Data Agendamento, Data Retirada, Cadastro (usuario), Status Locação. | |
| Disponibilizar uma API Rest que deverá disponibilizar o serviço de autenticação e autorização para usuários | Nesta fase avaliamos o domínio dos frameworks para uso de API Rest, Segurança, Documentação e uso da API desenvolvida |
| Disponibilizar uma API Rest com todas as funcionalidades citadas acima | |
| Disponibilizar a documentação dos recursos do Webservices REST. | |

Backend:

- Apresentar a estrutura do projeto no github;
- Apresentar a API do projeto devidamente documentada e preferencialmente no Heroku;
- Realizar uma demonstração de uma jornada de consumo dos recursos para criar usuário;
- Realização do Cadastro com Endereço e Cadastro de Livros.
- Realizar operações de locação de livros considerando as fases conforme requisitos.
- Realizar a consulta das locações por data de agendamento, data retirada, cadastro e status.
-
- **Requisitos Técnicos:** Implementação aplicando as boas práticas de programação, uso de ORM, Framework de persistência, segurança na API, integridade do banco de dados.