

Laboratório 6 – Serviços Web REST e Node.js Express

1 Exercícios

1. Acesse o serviço REST <https://reqres.in/> e escreva um pequeno programa que consuma o serviço. Mostre o acesso a URIs diferentes para as operações de GET/POST/PUT/PATCH/DELETE. Pelo menos uma das operações deve testar tanto o retorno com sucesso quanto com falha.

2. Retome o exercício do Laboratório 5 sobre o sistema de biblioteca. Projete uma API de Web Service REST para expor operações de negócio sobre esta base de dados da Biblioteca. Siga o seguinte exemplo:

Verbo: GET

URI: /livros/{id}

Entrada: identificador do livro como String

Retorno:

- 200 OK com objeto contendo os dados de um livro serializado em JSON
- 404 Not Found se o livro não existe
- 500 Internal Server Error no caso de falha de processamento do servidor

3. Implemente o Serviço Web projetado no exercício anterior. Procure seguir um modelo de implementação de acordo com o padrão MVC apresentado em sala de aula.

4. Acesse o repositório <https://github.com/Microsoft/TypeScript-Node-Starter> e observe a estrutura do projeto e de como o Express foi utilizado para disponibilizar páginas de um site. Responda:

- a) Como as páginas são manipuladas pelo lado-servidor antes de serem enviadas ao lado-cliente? Dica: a estrutura segue o chamado padrão de projeto MVC – Model View Controller onde as Views são a implementação do padrão de projeto Template View.
- b) Quais módulos de middleware foram utilizados e qual o propósito de cada um? Procure observar que as funcionalidades adicionais estão trabalhando como complementares ao framework Express.
- c) Abra o arquivo “package.json” e observe os scripts de *build* da aplicação. O que são TSLINT, SAAS e JEST?