

## Exercício - Modelo de Maturidade de Richardson

Nome: Guilherme Franco Batista

GRR20211603

Descreva:

1. O que é o RMM.
2. Quais são os níveis do RMM e quais as características de cada.
3. O que é HATEOAS.

1) O RMM define quão compatível com REST é a API, Richardson usou 3 fatores: URL(recursos), HTTP(métodos e retornos) e HATEOAS, com isso ele encontrou 4 categorias.

- **2) Nível 0: The Swamp of POX (Plain Old XML)**
  - Nome dos recursos não seguem qualquer padrão
  - São usados somente para invocar métodos remotos
  - É usado somente um ou dois métodos HTTP: Ex somente POST
- Exemplo:
  - Inserir pessoa: /salvarPessoa
  - Obter uma pessoa: GET/obterPessoa/1
- **Nível 1: Resources (recursos)**
  - URL não é só usada para invocar métodos: Identifica um recurso
  - Recursos são usados de forma eficiente
  - Ainda não usa os métodos/verbos do HTTP de forma correta: somente GET e POST
- Exemplo:
  - Recurso Pessoa com id 1: **/pessoa/1**
  - Se quiser fazer qualquer operação nesta pessoa, deve-se usar este recurso

- **Nível 2: HTTP Verbs(verbos/métodos HTTP)**

- Implementa o nível 1
- Os métodos HTTP são usados de forma correta, com suas características
- GET, POST, PUT, DELETE
- Exemplo: GET é seguro, pode ser executado em qualquer ordem e sempre retorna o mesmo resultado, portanto pode ser armazenado em cache
- Também usa os status code de retorno de forma correta
- Também usa os headers das mensagens, para autenticação, por exemplo
- Também usa os query parameters para configurar os retornos de alguma forma, como paginação, filtros, etc

- Exemplo:

- Obter a pessoa 1: **GET /pessoa/1**
  - Retorna 200 se OK, 404 se não existe
- Inserir uma pessoa: **POST /pessoa**
  - Retorna 201 se criado, 409 se a pessoa já existe

- **Nível 3: Hypermedia (HATEOAS)**

- Implementa o nível 1 e 2
- Implementa hipermídia nos retornos
- Os retornos indicam as URLs das operações que podem ser feitas a seguir

3) HATEOAS é um modelo para que uma API REST permita que aplicações que a consumam naveguem entre seus recursos através de links e URLs sem a necessidade de conhecimento denso prévio sobre a API.