



# **Entendendo e Documentando RESTful APIs**

# Modelo de maturidade Richardson

# Modelo de maturidade Richardson

---

“Apesar de Roy Fielding deixar bastante claro que para uma API ser considerada RESTful ela precisa obrigatoriamente seguir todas as constraints definidas em seu trabalho, na prática, muitas vezes precisamos de uma abordagem um pouco mais simples.”

# Modelo de maturidade Richardson

— — —

“Sendo assim, Leonard Richardson propôs um modelo de 4 níveis para que alcancemos uma API REST”

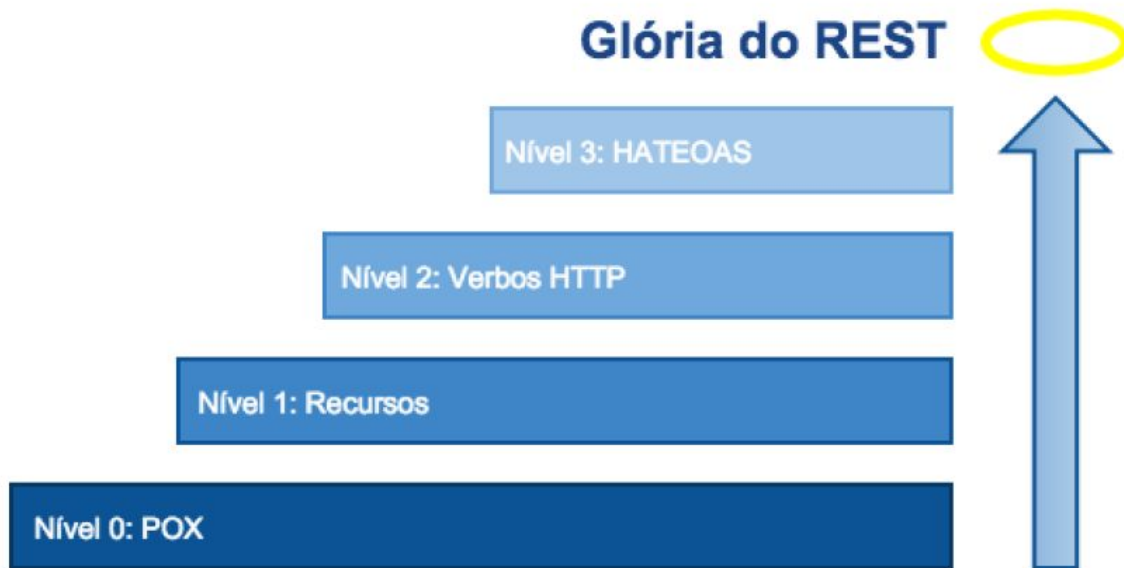
# Modelo de maturidade Richardson

— — —

“Os níveis 0, 1 e 2 talvez sejam mais familiares, e de fato são mais fáceis de implementar, porém, deve ficar claro que os mesmos não são considerados RESTful.”

# Modelo de maturidade Richardson

— — —



<http://martinfowler.com/articles/richardsonMaturityModel.html>

**Nível 0 - POX**

# Nível 0 - POX

— — —

“Você certamente já deve ter desenvolvido ou visto em algum lugar uma API que segue esse modelo de design. Apesar de ser o nível mais distante do que de fato REST propõe, muitas APIs ditas como RESTful se encontram neste nível de maturidade.”



# Nível 0 - POX

— — —

“Neste nível, as mensagens podem ser serializadas em formatos como XML, JSON ou outros. É importante lembrar, como dito anteriormente, que não é o formato da mensagem que define ou não um sistema REST.”

# Nível 0 - POX

— — —

POST **/salvarCliente**

<Cliente>

<Nome>Manoel da Silva</Nome>

...

</Cliente>

# Nível 0 - POX

---

RPC (POX)		
Verbo HTTP	URI	Ação
GET	/buscarCliente/1	Visualizar
POST	/salvarCliente	Criar
POST	/alterarCliente/1	Alterar
GET/POST	/deletarCliente/1	Remover

# Nível 0 - POX

---

REST		
Verbo HTTP	URI	Ação
GET	/cliente/1	Visualizar
POST	/cliente	Criar
PUT	/cliente/1	Alterar
DELETE	/cliente/1	Remover

# Nível 0 - POX

---

“Um outro problema constantemente encontrado, é a manipulação incorreta dos códigos de resposta do HTTP. Códigos e mensagens de erros são frequentemente manipuladas nas mensagens geradas pela aplicação, o que impede que elementos de gateway e proxy trabalhem de forma adequada.”

# Nível 0 - POX

---

GET /buscarCliente/1

**HTTP/1.1 200 OK**

<buscarCliente>

<status>CLIENTE NÃO ENCONTRADO</status>

<codigo>**404**</codigo>

<buscarCliente>

# Nível 0 - POX

— — —

“Apesar da mensagem sugerir que o cliente solicitado não foi encontrado, a resposta HTTP apresenta uma informação totalmente diferente (200 OK), ou seja, existe uma diferença semântica entre o protocolo HTTP e a representação gerada pela aplicação.”

# Nível 0 - POX

— — —

“Apesar da mensagem sugerir que o cliente solicitado não foi encontrado, a resposta HTTP apresenta uma informação totalmente diferente (200 OK), ou seja, existe uma diferença semântica entre o protocolo HTTP e a representação gerada pela aplicação.”



# Nível 0 - POX

---

Em resumo, no nível 0 podemos dizer que:

**“Você nem mesmo está usando o HTTP  
de forma correta!”**