

The background features a city skyline at dusk or dawn, with a body of water in the foreground. Overlaid on this is a network of white lines connecting various nodes. Several nodes are highlighted with red circular icons: a laptop, a smartphone, a tablet, and a stylized red 'R' logo. A large, stylized red 'C' made of a network of nodes is positioned to the right of the main text.

Embarcadero Conference

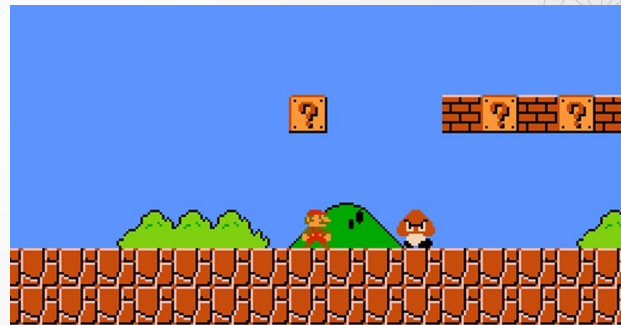


Integrando sua App ao Mundo via REST/JSON

Mário Guedes – jmarioguedes@gmail.com

Mário?

- Gerente de Desenvolvimento na Contact Studio Software
 - *Estamos contratando!*
 - <http://www.g4solutions.com.br/trabalhe-conosco/>
- Desenvolvedor Delphi, Python, JS e noSQL
- 15+ anos na lida
- Filho de Valdete e Joselito
- Irmão do Manoel, da Jenny e do Jonhy
- Pai do Júlio e da Fernanda
- E noivo da Tamires



Iniciando

- **REST:** Estilo arquitetural criado por Roy Fielding
- **RESTful:** Design de adoção que respeita os conceitos REST
- **REST API:** Dados e funções de negócios expostos via web para outras aplicações consumirem
- **JSON:** Representação de Objetos JavaScript
 - Formato em texto simples e auto descritivo
 - Permite encadeamento de dados
 - Permite intercâmbio de informação entre plataformas heterogêneas (Delphi X Python por exemplo)
 - Menos verboso que o XML
 - Criado por Douglas Crockford
 - <http://json.org/json-pt.html>

HTTP – Protocolo de Transferência de Hipertexto

- Entendendo o HTTP entenderemos o REST
- Baseado em texto: *Simplicidade*
- Orientado à documento
- Modelo de Requisição e Resposta: *Síncrono*
- Possui uma **ação**, um **endereçamento**, um **cabeçalho** e talvez um **conteúdo**

URL: Recurso a ser trabalho

- A URL identifica o recurso a ser trabalhado.
- Não importa o que será feito com o recurso, a URL será a mesma.
- O Verbo HTTP é que determinará o que será feito.
- As partes de uma URL determinam uma hierarquia em relação ao recurso.
- Para consumir um servidor DataSnap/REST usamos URL por justaposição.

Interface uniforme

- A ação a ser executada sobre um recurso é determinado pelo **verbo HTTP**.
- Os quatro verbos principais equivalem às operações CRUD.

	Verbo	Ação
Create	POST	Cria um novo recurso
Read	GET	Recupera um determinado recurso ou lista de recursos
Update	PUT	Atualiza um determinado recurso ou cria se inexistente
Delete	DELETE	Elimina um determinado recurso

Principais Status HTTP

- **200 – OK:** A solicitação foi processada com sucesso!
- **201 – CREATED:** A solicitação de criação de um recurso (POST) foi executada com sucesso!
- **202 – ACCEPTED:** A solicitação de execução de uma tarefa assíncrona foi enfileirada com sucesso!
- **304 – NOT MODIFIED:** O conteúdo armazenado em cache pela aplicação cliente é válida!
- **400 – BAD REQUEST:** A solicitação possui algum erro por parte da aplicação cliente.
- **401 – UNAUTHORIZED:** O usuário não pode executar a operação desejada.
- **404 – NOT FOUND:** Clássico! O recurso solicitado não existe.
- **500 – INTERNAL SERVER ERROR:** Ocorreu um erro na camada servidora.

JSON – Notação de Objeto JavaScript

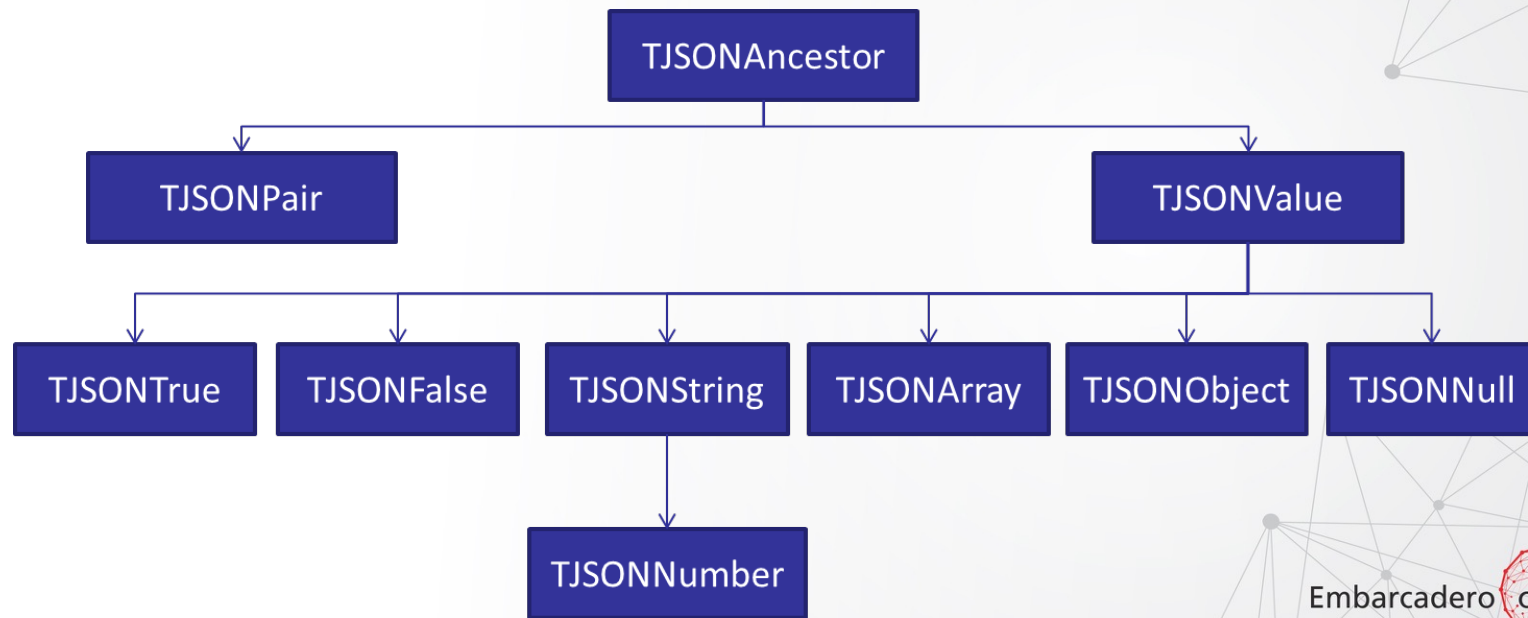
- Um recurso deve ser representado de alguma forma.
- Mas como podemos representar um vendedor ou uma empresa?
 - Precisamos de uma estrutura complexa.
- Poderia ser XML, INI, CSV, YAML... Mas a indústria adotou o JSON.
- Legibilidade!
- Maior compactação resulta em maior vazão.
- Aderência ao JavaScript: **A** linguagem frontend web.

Paleta REST Client

- **TRESTClient:** Responsável pela configuração de acesso à um serviço RESTful.
- **TRESTRequest:** Responsável pela requisição de um recurso específico.
- **TRESTResponse:** Responsável pela decodificação da resposta.
- HTTPS: <http://indy.fulgan.com/SSL/>

JSON é simples!

- A unit `System.JSON` possui várias classes para lidar com o JSON no Delphi.



API REST

- Grandes serviços web possuem uma REST API.
- É um conjunto de URLs e representações JSON bem documentadas.
- Facebook, Twitter, Google, LinkedIn, Uber, Spotify, Netflix, Instagram, Amazon, GitHub, ~~RedTube~~ e provavelmente o seu serviço web preferido possuiu APIs REST.
- Deve-se ler a documentação e ficar atento aos termos de uso. Em geral há uma área no site para desenvolvedores.
- E o seu sistema pode entrar neste time: DataSnap/REST.

Exemplo prático - Suas músicas do Spotify

- <https://developer.spotify.com/web-api/endpoint-reference/>
- OAuth 2.0
- Vamos listar nossos álbuns e playlists
- Vamos adicionar uma nova playlist ao nosso usuário

Para saber mais

- **REST & JSON, Chega de Dúvidas!**

Rodrigo Mourão

https://youtu.be/k7J_FJWiFu8?t=15m:45s

- **Integração de Serviços Cloud com REST/JSON**

Fernando Rizzato

<https://youtu.be/azFARYrYKR4>

- **Tudo sobre o REST Client Library**

Mário Guedes

<https://youtu.be/ajl2GEJonQA>

@ jmarioguedes@gmail.com

@ mario.guedes@contactstudio.com

 /jmarioguedes

 /jmarioguedes

[OBRIGADO]

Embarcadero



conference