

# Treinamento

# { **REST:API** } + Delphi

**Autor:** Gleryston Matos

**GitHub:** [GlerystonMatos](#)

**Medium:** [Gleryston Matos](#)

**Linkedin:** [Gleryston Matos](#)

## Introdução

Neste treinamento veremos o funcionamento básico de três tecnologias para desenvolver REST API com Delphi e diferentes formas de utilizá-las.

## DataSnap

DataSnap é uma tecnologia de software, desenvolvida pela embarcadero, para implementar sistemas distribuídos e aplicações multicamadas.

### Links:

[Versões do Delphi](#)

[O que é DataSnap](#)

[Configurar verbos HTTP](#)

[Problemas com PUT, DELETE IIS 8.5](#)

[Tutorial: DataSnap Application Using an ISAPI DLL Server \(IIS\)](#)

[Building, debugging, and deploying a DataSnap/REST ISAPI dll](#)

[DataSnap REST Application Wizard](#)

### Exemplos:

[rest-api-datasnap](#)

### 01 - Instalação:

Não necessita de instalação pois é nativo do delphi, porém apenas nas versões architect / enterprise.

### 02 - Tipos de projeto:

01 - Aplicação (Sistema Operacional).

02 - [ISAPI](#). (IIS)

### 03 - Explicação do projeto:

01 - Composição.

02 - Funcionamento.

#### 04 - Realização de testes:

- 01 - URL base.
- 02 - Endpoints.
- 03 - Verbos HTTP.

#### Observações:

- 01 - Não é muito complexo em sua implementação.
- 02 - Fácil de usar e configurar. (Depois de aprender)
- 03 - Muito conteúdo disponível na internet.
- 04 - Não aguenta grandes fluxos. (Problemas no gerenciamento de memória)

## Rest Dataware

O Rest Dataware é uma suíte de componentes que abstrai a complexidade para acesso a dados remotos em uma aplicação construída utilizando a arquitetura multicamadas totalmente free e de código fonte aberto.

#### Links:

[Apache](#)  
[REST Dataware](#)  
[Rest Dataware está incrível](#)

#### Vídeos:

[Criação e Execução de Eventos \(Métodos\) no Servidor](#)  
[Error: Apache Shutdown unexpectedly in Xampp 2016](#)  
[Conectando os componentes dbware do RDW](#)  
[Criando meu primeiro Servidor do ZERO](#)  
[Postman com Rest Dataware](#)  
[Apache CGI Server](#)

#### Exemplos:

[rest-api-restdataware](#)

#### 01 - Instalação:

No site oficial do componente existe um vídeo tutorial que demonstra a instalação do componente:  
[Instalação Manual do REST Dataware](#)

Para verificar a versão da release do delphi que será usada, basta verificar a seguinte [tabela](#).

**Observação:** Dependendo da versão do delphi, pode ser necessário adicionar mais diretórios ao library path.

#### 02 - Tipos de projeto:

- 01 - Aplicação (Sistema Operacional).
- 02 - [ISAPI](#). (IIS)
- 03 - Console CGI. (Apache)

### 03 - Explicação do projeto:

- 01 - Composição.
- 02 - Funcionamento.

### 04 - Realização de testes:

- 01 - URL base.
- 02 - Endpoints.
- 03 - Verbos HTTP.

### Observações:

- 01 - Fácil de usar e configurar. (Depois de aprender)
- 02 - Tem uma boa comunidade.
- 03 - É open source.
- 04 - Não tem muito conteúdo disponível na internet. (A maioria em vídeo, o que dificulta pesquisas)
- 05 - Feito para resolver os problemas do DataSnap. (Gerenciamento de memória entre outros).
- 06 - A maior parte da implementação é em componentes, podendo gerar problema durante o merge.
- 07 - Possui muitos recursos nativos não necessitando de outros componentes. (Autenticação e encriptação)

## Horse

Horse é um framework Web inspirado no Express para Delphi.  
Projetado para facilitar o desenvolvimento rápido de maneira minimalista e com alto desempenho.

### Links:

[Repositório do Horse](#)  
[Construindo APIs REST em Delphi + Horse](#)

### Vídeos:

[Horse: A revolução no REST](#)  
[Horse: criando uma API escalável e minimalista \(Vinicius Sanchez - Fiorilli Software\)](#)

### Exemplos:

[rest-api-horse](#)

### 01 - Instalação:

No vídeo [Horse: A revolução no REST](#) temos o passo a passo para realizar a instalação do Horse.

- 01 - Instalação do [boss](#). (SHIFT + Botão direito, para abrir powershell na pasta)
- 02 - Iniciar o boss na pasta do projeto. (boss init)
- 03 - Instalar o [horse](#) usando o boss. (boss.json)
- 04 - Instalar o [jhonson](#) usando o boss. (boss.json)

### 02 - Tipos de projeto:

- 01 - Aplicação. (Sistema Operacional)
- 02 - Console. (Sistema Operacional)

### 03 - Explicação do projeto:

- 01 - Composição.
- 02 - Funcionamento.

### 04 - Realização de testes:

- 01 - URL base.
- 02 - Endpoints.
- 03 - Verbos HTTP.

### Observações:

- 01 - Fácil de usar e configurar, por ser bastante intuitivo.
- 02 - Tem uma boa comunidade.
- 03 - É open source.
- 04 - Possui gerenciador de versão próprio para ele e para seus middlewares.
- 05 - Podemos criar middlewares personalizados para ele.
- 06 - Possui vários middlewares prontos para serem usados.
- 07 - Na página dos repositórios existem exemplos de como usar cada ferramenta.
- 08 - Não usa componentes, apenas código.
- 09 - Não tem muito conteúdo disponível na internet. (A maioria em vídeo, o que dificulta pesquisas)

### Executar testes no CMD:

- 01 - [Newman](#)
- 02 - Comandos:

```
newman run D:\Projetos/rest-api-restdataware/Postman/rest-api-delphi.postman_collection.json --environment D:\Projetos/rest-api-restdataware/Postman/rest-dataware-console-cgi-apache.postman_environment.json --insecure
newman run D:\Projetos/rest-api-datasnap/Postman/rest-api-delphi-datasnap.postman_collection.json --environment D:\Projetos/rest-api-datasnap/Postman/datasnap-isapi-iis.postman_environment.json --insecure
newman run D:\Projetos/rest-api-datasnap/Postman/rest-api-delphi-datasnap.postman_collection.json --environment D:\Projetos/rest-api-datasnap/Postman/datasnap-application.postman_environment.json --insecure
newman run D:\Projetos/rest-api-horse/Postman/rest-api-delphi.postman_collection.json --environment D:\Projetos/rest-api-horse/Postman/horse-console-application.postman_environment.json --insecure
newman run D:\Projetos/rest-api-horse/Postman/rest-api-delphi.postman_collection.json --environment D:\Projetos/rest-api-horse/Postman/horse-application.postman_environment.json --insecure
newman run D:\Projetos/rest-api-restdataware/Postman/rest-api-delphi.postman_collection.json --environment D:\Projetos/rest-api-restdataware/Postman/rest-dataware-application.postman_environment.json --insecure
newman run D:\Projetos/rest-api-restdataware/Postman/rest-api-delphi.postman_collection.json --environment D:\Projetos/rest-api-restdataware/Postman/rest-dataware-isapi-iis.postman_environment.json --insecure
```

### Comparação prática:

[Datasnap x Horse x REST Dataware](#)