

Centro de Pesquisas do Instituto Mauá de Tecnologia  
Divisão de Eletrônica e Telecomunicações

**IMT – ACESSO À REST API**  
***(NETWORK SERVER)***

São Caetano do Sul

2018



## RESUMO

A recuperação de dados enviados através dos *nodes* por cada aplicação é importante em relação ao histórico de dados disponível. Além de ser um backup dos dados caso uma conexão *MQTT broker* não seja estabelecida pela aplicação, permite -se que o usuário realize um *backup* de tempos em tempos para o próprio banco de dados. A saber que o banco de dados do *Network Server* apenas garante o armazenamento dos dados referentes aos últimos 7 dias.

Para que isso ocorra com segurança entre as camadas servidor e aplicação, foi desenvolvida uma *REST API* (*Representational State Transfer Application Programming Interface*) em que através de uma requisição *HTTP* (*HyperText Transfer Protocol*) do tipo *GET*, são retornados os últimos valores inseridos no banco de dados de maneira decrescente. Isto é, do mais atual ao mais antigo de acordo com o parâmetro de quantidade de *logs* inserido no momento da requisição. Esta documentação especifica os detalhes deste tipo de requisição e métodos de retorno. Para um exemplo de aplicação com este recurso deve-se consultar a documentação **EXEMPLO DE APLICAÇÃO REST-API (NODE-RED)**

**Palavras chave:** histórico, dados , *backup*, armazenamento, requisição, *GET*.

## SUMÁRIO

<b>1</b>	<b>REQUISITOS</b>	<b>5</b>
----------	-------------------	----------

## 1 REQUISITOS

Para se conectar à REST-API de maneira a recuperar os dados através do banco de dados do [Network Server – Instituto Mauá de Tecnologia](#), é necessário que a aplicação faça uma requisição *HTTP (Hypertext Transfer Protocol)* do tipo *GET*. É através desta requisição em determinada *URL (Uniform Resource Locator, isto é, endereço de recurso disponível)* que será permitida a recuperação dos dados para se realizar um gráfico de histórico, bem como uma gravação de backup para o banco de dados da própria aplicação.

Através da requisição, deverá ser realizado um comando do tipo *GET* na seguinte *URL*:

[https://networkserver.maua.br/api/index.php/{appSkey}/{#requisições}/{dev\\_EUI}](https://networkserver.maua.br/api/index.php/{appSkey}/{#requisições}/{dev_EUI})

Em que os parâmetros descritos devem ser preenchidos segundo o exemplo abaixo:

- *appSkey*: *string* de 32 *bytes* referente à chave da aplicação gerada na ativação do *node*. Neste caso, segundo a criação do *node* Hidrometro-01 na aplicação Hidrometros, a chave corresponde a “842dd04973596cec7864109a8335a455”
- *#requisições*: número (inteiro) de registros a serem recuperados do banco de dados em ordem decrescente.
- *dev\_EUI*: Número correspondente *ao end-device unique identifier* de cada *node* cadastrado no *Network Server*.

Como um exemplo de requisição para o *node* Hidrometro-01 de maneira a recuperar os últimos dois registros enviados do *node* ao servidor, tem-se:

<https://networkserver.maua.br/api/index.php/842dd04973596cec7864109a8335a455/2/0004a30b001a5ea2>

Como resultado desta requisição, obtem-se a seguinte resposta, no formato catacterístico *JSON*:

- logs:
  - 0:
    - created\_at: "2018-03-07 13:34:44.961006-03",
    - dev\_eui: "0004a30b001a5ea1",
    - fCnt: 196,
    - fPort: 100,
    - rx\_info\_set:
      - 0:
        - mac: "7276ff00080801db",
        - rssi: -108,
        - size: 20,
        - time: "0001-01-01T00:00:00Z",
        - channel: 3,
        - loRaSNR: 8.2,
        - rfChain: 0,
        - codeRate: "4/5",
        - dataRate:
          - bandwidth: 125,
          - modulation: "LORA",
          - spreadFactor: 7
        - crcStatus 1,
        - frequency 902900000,
        - timestamp 3308224083,
      - phy\_payload "40d242ba0680c4006468ff8328c3614d583b1500",
      - data\_payload "0b023d900c0cc9"

- logs:
  - 1:
    - created\_at: "2018-03-07 13:27:02.503976-03",
    - dev\_eui: "0004a30b001a5ea1",
    - fCnt: 195,
    - fPort: 100,
    - rx\_info\_set:
      - 0:
        - mac: "7276ff00080801db",
        - rssi: -107,
        - size: 20,
        - time: "0001-01-01T00:00:00Z",
        - channel: 4,
        - loRaSNR: 9,
        - rfChain: 1,
        - codeRate: "4/5",
        - dataRate:
          - bandwidth: 125,
          - modulation: "LORA",
          - spreadFactor: 7
        - crcStatus 1,
        - frequency 903100000,
        - timestamp 2845761723,
      - phy\_payload "40d242ba0680c300644a862cf61d7363c11a5de7,
      - data\_payload "0b023d7f0c0cc9"

Os campos úteis que trazem consigo informações do dado enviado pelo node são os campos `created_at` (data da criação do registro no servidor), `fCnt` (frame count, isto é, contador de frames enviados do node ao servidor) e `data_payload` em que efetivamente se tem o dado enviado pelo node. A este campo deve-se recuperar o dado equivalente de maneira idêntica ao enviado pelo node.

Para exemplo de aplicação Node-RED utilizando a requisição REST-API GET, pode-se ler a documentação *EXEMPLO DE APLICAÇÃO REST-API (NODE-RED)*.