



Protocolo de Integração de Dados Entre os Sistemas de Pesagem Instalados em Araranguá/SC

Agência Nacional de Transportes Terrestres – ANTT
Recurso de Desenvolvimento Tecnológico – RDT
**Estudos e pesquisas aplicadas visando à modernização da fiscalização e à monitoração
rodoviária com equipamentos de pesagem em alta velocidade utilizando o sítio de
pesquisas instalado em Araranguá/SC**

Protocolo de Integração de Dados Entre os Sistemas de Pesagem Instalados em Araranguá/SC

Março de 2023



Agência Nacional de Transportes Terrestres – ANTT / Recurso de Desenvolvimento Tecnológico – RDT

Estudos e pesquisas aplicadas visando à modernização da fiscalização e à monitoração rodoviária com equipamentos de pesagem em alta velocidade utilizando o sítio de pesquisas instalado em Araranguá/SC

Concessionária Catarinense de Rodovias S.A. – CCR Via Costeira

Rua Silvio Burigo, nº 555 – Bairro Oficinas
CEP 88702-500 – Tubarão – Santa Catarina
<https://www.ccrviacosteira.com.br>

Breno Ferreira Leal
Gerente Contrato de Concessão

Fundação de Amparo à Pesquisa e Extensão Universitária – FAPEU

Rua Delfino Conti, s/nº, Campus Universitário, Bairro Trindade – Caixa Postal 5153
8040-370 – Florianópolis – Santa Catarina
<http://www.fapeu.com.br>

Valter Zanela Tani
Coordenador Técnico
Fábio Silva de Souza
Superintendente

Lista de Figuras

Figura 1	Exemplo de objeto JSON	3
Figura 2	Exemplo de solicitação de autenticação e resposta obtida caso tenha sucesso	6
Figura 3	Exemplo de solicitação de autenticação e resposta obtida caso haja alguma falha	6
Figura 4	Exemplo de inserção de novos registros na base de dados.....	8
Figura 5	Exemplo de consulta de registros de pesagem.....	10
Figura 6	Exemplo de consulta de registro específico.....	11

Lista de Quadros

Quadro 1	Campos do objeto JSON para inserção de registros pela API	7
Quadro 2	Campos do objeto JSON para consulta de registros por meio da API	9

Sumário

1. Introdução	1
2. Protocolo de Integração Utilizado no Sítio de Pesquisas de Araranguá/SC.....	3
2.1. Formatação das informações trocadas entre os sistemas	3
2.2. Interface e protocolo de troca de informações.....	4
2.3. Utilização da interface de integração (API)	4
2.3.1. Autenticação.....	5
2.3.2. Adição de novos registros.....	6
2.3.3. Consulta de registros do banco de dados	9
3. Considerações Finais	13
REFERÊNCIAS	15

1. Introdução

O sítio de pesquisas de pesagem em movimento de Araranguá/SC realiza a pesquisa e o teste conjunto de diversas tecnologias relacionadas à fiscalização de peso e à monitoração rodoviária automatizadas. O tratamento e a análise dos dados coletados pelos diversos sistemas instalados são feitos pela equipe de pesquisa, responsável pela manutenção do sítio. O “Protocolo de Integração de Dados entre os Sistemas de Pesagem Instalados em Araranguá/SC”, Produto 5 do projeto de RDT intitulado “Estudos e Pesquisas Aplicadas Visando à Modernização da Fiscalização e Monitoração Rodoviária com Equipamentos de Pesagem em Alta Velocidade Utilizando o Sítio de Pesquisas Instalado em Araranguá/SC”, apresentado neste documento, é uma ferramenta por meio da qual a equipe de pesquisa pode integrar os dados dos diversos parceiros e tecnologias que estão em teste no sítio de pesquisa. O protocolo visa, também, permitir que sejam instalados mais sistemas e tecnologias para teste no sítio de pesquisas, com maior facilidade na integração dos dados entre os sistemas.

Um protocolo de integração é um conjunto de regras e procedimentos que definem como diferentes sistemas, aplicativos e dispositivos de hardware se comunicam e interagem entre si. Ele é usado para permitir a interoperabilidade entre diferentes sistemas, permitindo que eles trabalhem juntos, de forma harmoniosa, para atingir um objetivo comum. Os protocolos de integração são geralmente projetados para suportar a troca de informações entre sistemas de diferentes plataformas, tecnologias e fabricantes. Eles são usados em uma ampla variedade de contextos, incluindo integração de sistemas empresariais, integração de dispositivos em rede, integração de sistemas de automação industrial e integração de sistemas de saúde.

Os protocolos de integração podem incluir especificações para formatos de dados, mecanismos de autenticação e autorização, gerenciamento de erros, fluxo de trabalho e outras funções relacionadas à comunicação e à interoperabilidade. Alguns exemplos comuns de protocolos de integração incluem:

- *Representational State Transfer (REST)*: um protocolo de integração baseado em *Hypertext Transfer Protocol (HTTP)*, que permite a comunicação entre sistemas de computador, usando recursos e representações de recursos, que são mapeados em *Uniform Resource Locators (URLs)*. Ele é amplamente utilizado para integração de serviços web e aplicativos móveis.
- *Simple Object Access Protocol (SOAP)*: um protocolo de integração baseado em *Extensible Markup Language (XML)*, que permite que diferentes sistemas se

comuniquem usando mensagens codificadas em XML. Ele é amplamente utilizado para integração de sistemas empresariais e aplicativos de negócios.

- *Message Queuing Telemetry Transport (MQTT)*: um protocolo de integração leve projetado para comunicação entre dispositivos de Internet das Coisas (IoT). Ele usa um modelo de publicação/assinatura para permitir que os dispositivos enviem e recebam mensagens.

A escolha do protocolo de integração depende dos requisitos específicos de cada aplicação, incluindo o tipo de sistemas que estão sendo integrados, as restrições de equipamentos e rede e as necessidades de segurança, confiabilidade e desempenho.

2. Protocolo de Integração Utilizado no Sítio de Pesquisas de Araranguá/SC

Levando em consideração a estrutura computacional atualmente existente no sítio de pesquisas, foi definido um protocolo de comunicação, efetivado através de objetos formatados pela notação de objetos *JavaScript Object Notation* (JSON) que são trocados por meio de uma interface de programação de aplicativos de *Application Programming Interface* (API), com protocolo REST.

2.1. Formatação das informações trocadas entre os sistemas

O formato utilizado para a troca de informações entre os sistemas é o formato JSON. É um formato de dados baseado em texto, seguindo a sintaxe de objetos *JavaScript*, que é de fácil leitura para o ser humano e de fácil interpretação para os sistemas. Utiliza convenções que são familiares em diversas linguagens de programação, facilitando a integração entre sistemas implementados com tecnologias diferentes (JSON, [20--]).

A sintaxe é composta por estruturas chamadas de “campos”, cada um composto por uma chave e seu respectivo valor. No objeto em formato JSON de exemplo, apresentado na Figura 1, o campo “*name*” possui valor “Marcelo da Silva”, o campo “*idade*” possui o valor “28”, e assim por diante.

Figura 1 – Exemplo de objeto JSON

```
{
  "email": "marcelo@empresa.com",
  "name": "Marcelo da Silva",
  "idade": 28,
  "altura": 1.72,
  "casado": True,
}
```

Fonte: FAPEU (2021)

2.2. Interface e protocolo de troca de informações

Os objetos, formatados através do formato JSON, são trocados entre os sistemas para realizar a comunicação e a integração dos dados. Para isso, um protocolo de comunicação é definido e estabelecido por meio de uma API (AWS, [20--]). A definição do protocolo inclui quais campos e quais valores devem ser enviados, dentro dos objetos JSON, para efetivar um tipo de troca entre os sistemas.

As API são mecanismos que permitem que dois componentes de software se comuniquem usando um conjunto de definições e protocolos. Geralmente, uma API é explicada em termos de “cliente” e “servidor”, nos quais a aplicação que envia uma solicitação é chamada de “cliente” e a aplicação que responde é chamada de “servidor”.

A API pode ser compreendida como um protocolo de comunicação estabelecido entre os sistemas que realizam a coleta dos dados e o servidor de banco de dados unificado. Ela define qual é o formato através do qual os sistemas integradores devem enviar as informações coletadas, quais informações devem estar contidas e para qual endereço da rede as informações devem ser enviadas. O servidor de API, portanto, fica disponível para receber as mensagens dos sistemas, contendo os registros de pesagem por eles coletados. Ao receber uma mensagem com dados de pesagem, o servidor de API realiza uma verificação básica dos dados enviados para garantir que são válidos e, em caso positivo, envia os dados para um banco de dados unificado da pesquisa.

No tipo de API adotado neste protocolo, os clientes e o servidor trocam dados utilizando o HTTP (W3C, 2014), que é composto por funções GET, POST, DELETE, dentre outras. A comunicação entre o cliente e o servidor é realizada por meio de endereços específicos de acesso, cada um relacionado a uma função específica.

2.3. Utilização da interface de integração (API)

A API de integração do sítio de pesquisas de Araranguá possui dois endereços de acesso, um para a autenticação e validação das informações de usuário e outro utilizado para inserir e consultar registros de pesagem no banco de dados unificado. Os endereços são disponibilizados somente para a rede interna do sítio de pesquisa, com acessos diretos configurados para cada sistema em teste:

- *http://endereçoapi:5000/v1/authenticate*: endereço para autenticação e solicitação de *token*.
- *http://endereçoapi:5000/v1/weighing*: endereço para cadastro e consultas de dados de pesagem.

A utilização da API desenvolvida é bastante simples, bastando fazer uma requisição HTTP para o endereço de acesso desejado, seja para cadastrar um novo registro, seja para fazer uma consulta na base de dados.

Por segurança, só terão permissão para utilizar a API usuários que estiverem devidamente cadastrados no sistema. O cadastro é gerenciado pela equipe de pesquisa do projeto de RDT (Recurso de Desenvolvimento Tecnológico), e os sistemas que integram o sítio possuirão informações específicas de autenticação. As informações deverão ser comunicadas durante as interações.

2.3.1. Autenticação

Para utilizar a API, primeiramente, o usuário necessita solicitar, ao servidor da API, um *token* de autenticação, que é obtido a partir do envio de uma requisição para um endereço de acesso de autenticação, informando seus dados de usuário e senha. O *token* recebido precisa ser enviado no momento em que o usuário enviar uma requisição de cadastro ou de consulta de dados. Por questões de segurança, o *token* tem uma validade de duas horas, sendo necessário que, depois desse período, seja solicitado um novo *token* para a API. A autenticação utilizada para a solicitação de *token* é do tipo HTTP básica.

Para iniciar o processo de autenticação, é necessário fazer uma requisição do tipo POST (W3C, 2014) para o endereço “<http://endereçoapi:5000/v1/authenticate>”. A requisição deve ter um cabeçalho de autorização intitulado “*Authorization*”, que contém a palavra “*Basic*” seguida de um espaço e de um nome de usuário e senha em texto, separados por dois pontos (:), utilizando a codificação “*base64*”.

Para fazer uma requisição com o usuário “teste” e com a senha “senha123456”, por exemplo, deve-se enviar, na requisição, o conteúdo “*Authorization: Basic dGVzdGU6c2VuaGExMjMONTY=*”, em que o trecho “*dGVzdGU6c2VuaGExMjMONTY=*” corresponde ao texto “teste:senha123456”, codificado em *base64*.

Na Figura 2, é exibido um exemplo de como fazer uma requisição para solicitar o *token* de acesso do usuário “teste” com senha “senha123456”. Na resposta, formatada em JSON, o campo “*status_code*” possui o valor 200, significando que o usuário e a senha informados estão corretos. Além disso, o *token* é retornado em formato JSON, acessível pelo campo “*access_token*”, cujo valor deverá ser guardado pelo sistema e adicionado no cabeçalho das requisições de cadastro e consulta de registros, que será demonstrado nos capítulos seguintes.

Quadro 1 – Campos do objeto JSON para inserção de registros pela API

Chave	Descrição	Exemplo de valor
datetime	Data e hora do evento de pesagem, no formato “ano-mês-dia hora:minutos:segundos”	2022-09-01 15:16:06
system_id	Código do sistema de pesagem, fornecido pela equipe de pesquisa	1
license_plate	Placa do veículo	ABC1C34
speed	Velocidade do veículo no momento da pesagem	55
pavement_temperature	Temperatura do pavimento	25
num_axles	Número de eixos do veículo	3
class	Classe do veículo de acordo com o Quadro de Fabricantes de Veículo (QFV) do Departamento Nacional de Infraestrutura de Transportes (DNIT)	3C
gross_weight	Peso total do veículo em kg	12465
weight_axle1	Peso do primeiro eixo do veículo em kg	4563
weight_axle2	Peso do segundo eixo do veículo em kg	5429
weight_axle3	Peso do terceiro eixo do veículo em kg	4335
weight_axle4	Peso do quarto eixo do veículo em kg	0
weight_axle5	Peso do quinto eixo do veículo em kg	0
weight_axle6	Peso do sexto eixo do veículo em kg	0
weight_axle7	Peso do sétimo eixo do veículo em kg	0
weight_axle8	Peso do oitavo eixo do veículo em kg	0
weight_axle9	Peso do nono eixo do veículo em kg	0
image	Imagem do veículo convertido para <i>base64</i>	

Fonte: FAPEU (2023)

Com um processo semelhante ao de solicitação de *token*, o cliente envia uma solicitação HTTP com o cabeçalho de autorização “*Authorization*” (que, neste caso, contém a palavra “*Bearer*” seguida de um espaço, conforme mostra a linha 10 da Figura 4) e o *token* recebido anteriormente, que deve estar válido. Na Figura 4, apresenta-se um exemplo de como realizar a inserção de um novo registro na base de dados através da API, escrito na linguagem de programação *Python*.

Figura 4 – Exemplo de inserção de novos registros na base de dados

```

import base64
import json
import requests
from requests.auth import HTTPBasicAuth

# Realiza o login e recebe o token.
def get_headers():
    url = 'http://localhost:5000/v1/authenticate'
    response = requests.post(url, auth=HTTPBasicAuth('usuario', 'senha123456'))
    token = response.json()['access_token']
    return {'Authorization': 'Bearer ' + token}

# Converte a imagem em base64.
def get_image(src_file):
    with open(src_file, "rb") as f:
        im_bytes = f.read()

    return base64.b64encode(im_bytes).decode("utf8")

# Realiza o request para inserir um registro
url = 'http://localhost:5000/v1/weighing'

data = {
    "datetime": "2022-09-01 15:16:06",
    "system_id": 1,
    "license_plate": "ABC1C23",
    "speed": 55,
    "pavement_temperature": 25,
    "num_axles": 3,
    "class": "3C",
    "gross_weight": 12543,
    "weight_axle1": 4563,
    "weight_axle2": 5329,
    "weight_axle3": 4335,
    "weight_axle4": 0,
    "weight_axle5": 0,
    "weight_axle6": 0,
    "weight_axle7": 0,
    "weight_axle8": 0,
    "weight_axle9": 0,
}

data['image'] = get_image(r'C:\veiculo.jpg')
response = requests.post(url, json = data, headers=get_headers())

#resposta do request
print(response)
#resposta do objeto JSON
print(response.json())

<Response [201]>
{"msg": "successfully registered", "weighing_id": 29172008, "datetime": "2022-09-01T15:16:06+00:00"}

```

Fonte: FAPEU (2023)

O *status_code* retornado, nesse exemplo, possui valor 201, significando que a inserção obteve sucesso. O retorno da requisição é composto, também, por um objeto JSON com dados de confirmação da inserção. Caso algum erro ocorra na solicitação para cadastro, a API retornará o *status_code* 500 (os códigos de retorno das requisições seguem o padrão de códigos de status do HTTP, (W3C, 2014)).

2.3.3. Consulta de registros do banco de dados

O processo para a realização de consultas é semelhante ao de cadastro de registros, utilizando o mesmo endereço, porém, com uma requisição do tipo GET e enviando, como parâmetro, um objeto JSON com os filtros desejados para a consulta.

No Quadro 2, está apresentada a descrição dos campos do objeto JSON utilizados para a consulta de registros.

Quadro 2 – Campos do objeto JSON para consulta de registros por meio da API

Chave	Descrição	Exemplo de valor
date_start	Data e hora inicial do evento de pesagem no seguinte formato “ano-mês-dia hora:minutos:segundos”	2022-09-01 15:16:06
date_final	Data e hora final do evento de pesagem no seguinte formato “ano-mês-dia hora:minutos:segundos”	2022-09-01 15:20:00
system_id	Código do sistema de pesagem, fornecido pela equipe de pesquisa	1
license_plate	Placa do veículo	ABC1C34
speed	Velocidade do veículo no momento da pesagem	55
	Temperatura do pavimento	25
num_axles	Número de eixos do veículo	3
Class	Classe do veículo de acordo com o QVF do DNIT	3C

Fonte: FAPEU (2023)

Apenas os campos “*date_start*” e “*date_final*” são obrigatórios, sendo os outros campos opcionais para a consulta. Na Figura 5, apresenta-se um exemplo, na linguagem de programação *Python*, de como fazer uma requisição de consulta dos dados para a API.

Figura 5 – Exemplo de consulta de registros de pesagem

```

import json
import requests
from requests.auth import HTTPBasicAuth

# Realiza o login e recebe o token.
def get_headers():
    url = 'http://localhost:5000/v1/authenticate'
    response = requests.post(url, auth=HTTPBasicAuth('usuario', 'senha123456'))
    token = response.json()['access_token']
    return {'Authorization': 'Bearer ' + token}

# Realiza o request para consultar um registro
url = 'http://localhost:5000/v1/weighing'

data = {
    "date_start": '2022-09-01 15:16:06',
    "date_final": '2022-09-01 15:20:00',
    "license_plate": "ABC1C23",
    "system_id": 1,
    "num_axles": 3,
    "class": "3C"
}

response = requests.get(url, json = data, headers=get_headers())

#resposta do request
print(response)
#resposta do objeto JSON
print(response.json())

```

```

<Response [201]>
{'data': {'class_id': None, 'datetime': '2022-09-01T15:16:06+00:00', 'dist_axle
1': None, 'dist_axle10': None, 'dist_axle11': None, 'dist_axle2': None, 'dist_a
xle3': None, 'dist_axle4': None, 'dist_axle5': None, 'dist_axle6': None, 'dist_
axle7': None, 'dist_axle8': None, 'dist_axle9': None, 'external_id': None, 'gro
ss_weight': 12543.0, 'lateral_position': None, 'license_plate': 'ABC1C23', 'num
_axles': 3, 'pavement_temperature': 25.0, 'speed': 55.0, 'system_id': 1, 'track
ing_id': None, 'vehicle_hash': None, 'weighing_id': 29172005, 'weight_axle1': 4
563.0, 'weight_axle10': None, 'weight_axle11': None, 'weight_axle12': None, 'we
ight_axle2': 5329.0, 'weight_axle3': 4335.0, 'weight_axle4': 0.0, 'weight_axle
5': 0.0, 'weight_axle6': 0.0, 'weight_axle7': 0.0, 'weight_axle8': 0.0, 'weight
_axle9': 0.0, 'weight_group1': None, 'weight_group10': None, 'weight_group2': N
one, 'weight_group3': None, 'weight_group4': None, 'weight_group5': None, 'weig
ht_group6': None, 'weight_group7': None, 'weight_group8': None, 'weight_group
9': None}, 'msg': 'successfully fetched'}

```

Fonte: FAPEU (2023)

Caso já exista o ID do registro de pesagem, cadastrado na base de dados unificada, é possível fazer uma requisição direta com esse número de ID informado, logo após o endereço de acesso. Um exemplo de como realizar esse procedimento é exibido na Figura 6.

Figura 6 – Exemplo de consulta de registro específico

```

import json
import requests
from requests.auth import HTTPBasicAuth

def get_headers():
    url = 'http://localhost:5000/v1/authenticate'
    response = requests.post(url, auth=HTTPBasicAuth('usuario', 'senha123456'))
    token = response.json()['access_token']
    return {'Authorization': 'Bearer ' + token}

url = 'http://localhost:5000/v1/weighing/29172005'

response = requests.get(url, headers=get_headers())

#resposta do request
print(response)
#resposta do objeto JSON
print(response.json())
    
```

```

<Response [201]>
{'data': {'class_id': None, 'datetime': '2022-09-01T15:16:06+00:00', 'dist_axle
1': None, 'dist_axle10': None, 'dist_axle11': None, 'dist_axle2': None, 'dist_a
xle3': None, 'dist_axle4': None, 'dist_axle5': None, 'dist_axle6': None, 'dist_
axle7': None, 'dist_axle8': None, 'dist_axle9': None, 'external_id': None, 'gro
ss_weight': 12543.0, 'lateral_position': None, 'license_plate': 'ABC1C23', 'num
_axles': 3, 'pavement_temperature': 25.0, 'speed': 55.0, 'system_id': 1, 'track
ing_id': None, 'vehicle_hash': None, 'weighing_id': 29172005, 'weight_axle1': 4
563.0, 'weight_axle10': None, 'weight_axle11': None, 'weight_axle12': None, 'we
ight_axle2': 5329.0, 'weight_axle3': 4335.0, 'weight_axle4': 0.0, 'weight_axle
5': 0.0, 'weight_axle6': 0.0, 'weight_axle7': 0.0, 'weight_axle8': 0.0, 'weight
_axle9': 0.0, 'weight_group1': None, 'weight_group10': None, 'weight_group2': N
one, 'weight_group3': None, 'weight_group4': None, 'weight_group5': None, 'weig
ht_group6': None, 'weight_group7': None, 'weight_group8': None, 'weight_group
9': None}, 'msg': 'successfully fetched'}
    
```

Fonte: FAPEU (2023)

Caso a consulta encontre dados de acordo com os filtros aplicados, nas duas formas de consulta, os dados serão retornados em um objeto JSON.

3. Considerações Finais

Os protocolos de integração são um elemento-chave para a interconexão de diferentes sistemas e dispositivos em diversas aplicações. É o caso do sítio de pesquisas de tecnologias de pesagem em alta velocidade, em Araranguá/SC, no qual diferentes sistemas, com diferentes tecnologias e fabricantes, funcionam lado a lado e geram dados que devem ser coletados e gerenciados pela equipe de pesquisa. O potencial da pesquisa desenvolvida no local está diretamente relacionado com a gestão das informações coletadas pelos diversos sistemas. Por este motivo, o “Protocolo de Integração de Dados Entre os Sistemas de Pesagem Instalados em Araranguá/SC”, proposto neste documento, visa facilitar a comunicação entre os sistemas e permitir que as informações sejam integradas de forma automatizada.

As informações integradas em um sistema unificado podem ser utilizadas para o desenvolvimento de pesquisas e análises, que são os objetivos deste projeto de RDT, visando ao avanço das tecnologias de pesagem em alta velocidade, e a exploração da possibilidade do seu uso para a fiscalização direta de peso.

REFERÊNCIAS

JSON. **Introdução ao JSON**. [20--]. Disponível em: <https://www.json.org/json-pt.html>. Acesso em: 09 mar. 2023.

W3C Architecture Domain. **HTTP – Hypertext Transfer Protocol**. 2014. Disponível em: <https://www.w3.org/Protocols>. Acesso em: 09 mar. 2023.

AWS. **O que é uma API?** [20--]. Disponível em <https://aws.amazon.com/pt/what-is/api>. Acesso em: 09 mar. 2023.



CCR *ViaCosteira*

CCR Via Costeira
Rua Silvio Burigo, 555 - Oficinas
CEP. 88702-500 - Tubarão - SC
0800-255-5550 - ouvidoria.viacosteira@grupoccr.com.br