

Universidade Federal de Goiás

Documentação da API do sistema GestVet

Estevão Cristino
Hyago Vieira de Souza
João Pedro Rodrigues Nunes Pinheiro
Pedro Henrique Coimbra e Silva

Goiânia, 28 de junho de 2019

1. Introdução

Este documento tem por finalidade descrever e documentar de forma completa a API do sistema de software GesteVet. Onde o sistema GesteVet faz o gerenciamento de dados para uma clínica veterinária.

Para isso o sistema tem disponível duas classes de persistência de dados que fazem a abstração de duas entidades principais para clínicas veterinárias, sendo elas a entidade Animal e Medicamento.

Para as requisições de Animal e Medicamento é necessário passar no header da requisição o token de acesso, onde este token é do tipo Bearer Token.

Com isso, o token de acesso é obtido através da requisição 2.1 Token. Onde, o usuário que deve ser passado no body é criado pela equipe e gerado para cada pessoa que queira acessar a API.

2. API

2.1 Token

- a. **POST** <https://gestvet.appspot.com/gestvet/auth/signin>

Requisição para gerar o token de acesso

Body: Objeto user que será analisado para gerar o token

Exemplo:

```
{
  "username": "hyagosouzza",      (String)
  "password": "123456",           (String)
}
```

Resposta:

200: Retorno com sucesso

Exemplo:

```
{
  "accessToken":
    "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJoeWFnb3NvdXp6YSIsImhhdCI6MTU2MjAy
    OTk0NCwiZXhwIjoxNTYyMTE2MzQ0fQ.BsxzwvHQvQMWKD3q2UCf2ZPeOjSZ
    P7iMe5v4kx9pnt4eJEVgqBAli2VrIC6hjCbUa_Q3wgq715MuSckhhuwfDg",
  (String)
  "tokenType": "Bearer",          (String)
}
```

400: Requisição inválida

3.1 Animal

- a. **POST** <https://gestvet.appspot.com/gestvet/animal/>

Adição de novo animal no sistema

Body: Objeto animal que será adicionado

Exemplo:

```
{
  "name": "Thor",          (String)
  "breed": "Vira-Lata",    (String)
  "species": "Cachorro",  (String)
  "age": 5                 (int)
}
```

Resposta:

200: Retorno com sucesso

400: Requisição inválida

b. **PUT** <https://gestvet.appspot.com/gestvet/animal/{id}>

Atualização de um animal no sistema

Id: Número de id do animal que será atualizado

Body: Objeto animal com as informações atualizadas

Exemplo:

```
{
  "name": "Amora",        (String)
  "breed": "Shih-tzu",    (String)
  "species": "Cachorro",  (String)
  "age": 4                (int)
}
```

Resposta:

200: Retorno com sucesso

400: Requisição inválida

c. **GET** <https://gestvet.appspot.com/gestvet/animals/>

Retorna a lista em array com todos os animais do sistema

Resposta:

200: Retorno com sucesso

Exemplo:

```
[
  {
    "name": "Amora",        (String)
    "breed": "Shih-tzu",    (String)
    "species": "Cachorro",  (String)
  }
]
```

```

        "age": 4                (int)
      },
      {
        "name": "Thor",        (String)
        "breed": "Vira-Lata",   (String)
        "species": "Cachorro", (String)
        "age": 5                (int)
      }
    ]

```

400: Requisição inválida

- d. **GET** <https://gestvet.appspot.com/gestvet/animal/{id}>
Retorna o animal referente a id passada na requisição

Id: Número de id do animal que será atualizado

Resposta:

200: Retorno com sucesso

Exemplo:

```

    {
      "name": "Amora",          (String)
      "breed": "Shih-tzu",      (String)
      "species": "Cachorro",    (String)
      "age": 4                  (int)
    }

```

400: Requisição inválida

- e. **DELETE** <https://gestvet.appspot.com/gestvet/animal/{id}>
Deleta o animal referente a id passada na requisição

Id: Número de id do animal que será atualizado

Resposta:

200: Retorno com sucesso

400: Requisição inválida

4.1 Medicamento

- a. **POST** <https://gestvet.appspot.com/gestvet/drug/>
Adição de novo medicamento no sistema

Body: Objeto medicamento que será adicionado

Exemplo:

```

{
  "name": "Bravekito",          (String)
  "price": 80,                  (double)
  "dosagem": "88ml",           (String)
  "quantity": 5                 (int)
}

```

Resposta:

200: Retorno com sucesso

400: Requisição inválida

b. PUT <https://gestvet.appspot.com/gestvet/drug/{id}>

Atualização de um medicamento no sistema

Id: Número de id do medicamento que será atualizado

Body: Objeto medicamento com as informações atualizadas

Exemplo:

```

{
  "name": "Frontline",          (String)
  "price": 90,                  (double)
  "dosagem": "120ml",           (String)
  "quantity": 3                 (int)
}

```

Resposta:

200: Retorno com sucesso

400: Requisição inválida

c. GET <https://gestvet.appspot.com/gestvet/drugs/>

Retorna a lista em array com todos os medicamentos do sistema

Resposta:

200: Retorno com sucesso

Exemplo:

```

[
  {
    "name": "Bravekito",          (String)
    "price": 80,                  (double)
    "dosagem": "88ml",           (String)
    "quantity": 5                 (int)
  },
  {
    "name": "Frontline",          (String)

```

```
        "price": 90,                (double)
        "dosagem": "120ml",         (String)
        "quantity": 3               (int)
    }
]
```

400: Requisição inválida

d. **GET** <https://gestvet.appspot.com/gestvet/drug/{id}>

Retorna o medicamento referente a id passada na requisição

Id: Número de id do medicamento que será atualizado

Resposta:

200: Retorno com sucesso

Exemplo:

```
{
  "name": "Bravekito",             (String)
  "price": 80,                     (double)
  "dosagem": "88ml",              (String)
  "quantity": 5                    (int)
}
```

400: Requisição inválida

e. **DELETE** <https://gestvet.appspot.com/gestvet/drug/{id}>

Deleta o medicamento referente a id passada na requisição

Id: Número de id do medicamento que será atualizado

Resposta:

200: Retorno com sucesso

400: Requisição inválida

3. Conexões do Projeto

a. **Conexão do Banco de Dados**

Banco de dados postgre hospedado na amazon, onde os dados são:

url - instance.cqdmal4ko4rf.us-east-1.rds.amazonaws.com

porta - 5432

nome - gestvetintegration

username - postgres

password - hyago0123

b. **Hospedagem do back-end**

O projeto em java que consiste no back-end da aplicação está hospedado no google app engine. Onde o domínio disponibilizado para a API é gestvet.appspot.com.

c. Hospedagem do front-end

O projeto Angular que consiste no front-end da aplicação está hospedado no infinite free. Onde o domínio disponibiliza <http://gestvet.epizy.com>