

Servidor API Olho Vivo

Implementação

Implementado usando a biblioteca Django 3.0.8 da linguagem Python e usando o banco de dados SQLite. Todos os requests são do tipo POST e GET, pois DELETE e PUT têm implementação pouco intuitiva.

Objetos

O servidor é responsável por armazenar quatro tipos de objetos que sintetizam a mecânica dos transportes. Eles são:

- Parada: contém informações sobre paradas pela cidade. Seus atributos são Latitude, Longitude (referentes à posição geográfica), Name e Id. A chave é Id.
- Linha: contém informações sobre linhas (trajetos). Seus atributos são Name, Id e um conjunto de paradas. A chave é Id.
- Veículo: contém informações sobre os veículos. Seus atributos são Name, Id, Modelo e LinhalId (referente ao Id da linha à que ele está vinculado). A chave é Id.
- PosiçãoVeículo: contém informações sobre a posição atual dos veículos. Seus atributos são Latitude, Longitude e Veiculoid (referente ao Id do veículo). A chave é Veiculoid.

Além dos quatro objetos principais, o servidor armazenará uma pequena lista de chaves de acesso, para evitar que usuários não autorizados utilizem os métodos POST. Essa lista só pode ser editada pela página de administrador do Django.

Requests (links)

É possível fazer consultas ou atualizações através de POST e GET requests. Para realizar GET, basta digitar algum dos links a seguir em algum navegador de internet. Já para realizar POST, é necessário o uso de softwares como o Postman (também é possível fazer GET por ele) e é necessário passar os parâmetros no corpo do requerimento, no formato x-www-form-urlencoded ou form-data, além de passar uma chave de autorização como parâmetro 'auth'.

Cada objeto possui sete métodos referentes ao seu CRUD e os links de cada um são semelhantes, basta substituir a palavra OBJETO pelo nome adequado: os nomes utilizados para cada objeto nos links são: Paradas, Linhas, Veiculos, Posicoes.

- Método de leitura geral: tipo GET que retorna uma relação (Id, Name) para cada item da tabela do objeto, no caso dos três primeiros tipos, ou (id, Veiculoid), no caso do último. O link para acesso é: URL_SERVIDOR/OBJETO.
- Método de leitura específica: tipo GET que retorna todas as características de um item da tabela do objeto, com base num id fornecido no link. Caso não haja um item com o id fornecido, retorna uma mensagem de erro explicativa. O link para acesso é URL_SERVIDOR/OBJETO/id
- Método de inserção: tipo POST que tenta inserir um novo item na tabela do objeto. Todos os atributos do item devem ser passados no corpo do requerimento, no formato e devem ser identificados com o nome exato definido no objeto. Caso os atributos sejam válidos e a inserção não cause repetimento de chave, o item novo é inserido e uma mensagem de sucesso é retornada. Caso contrário, retorna uma mensagem de erro explicativa. O link para acesso é URL_SERVIDOR/OBJETO/new
- Método de atualização: similar ao de inserção, exceto que não é necessário fornecer todos os atributos e só é possível utilizá-lo passando o id de um item válido da tabela do objeto. Somente os atributos passados serão atualizados. Não é permitido alterar o Id do item.

Qualquer erro que ocorra é retornado em mensagem explicativa. Caso o objeto seja do tipo Linha e seja fornecida uma (ou mais) parada, ela será adicionada à lista de paradas do item. O link para acesso é URL_SERVIDOR/OBJETO/update/id

- Método de remoção: tipo POST que tenta remover um item da tabela do objeto, dado um id fornecido. Se o item com o id existir, ele é removido. Caso contrário, retorna uma mensagem de erro explicativa. O link para acesso é URL_SERVIDOR/OBJETO/delete/id
- Método de importar novos objetos: tipo POST que recebe um arquivo csv passado como parâmetro 'file' que pode conter várias linhas, sendo que cada uma contém todos os campos de um objeto do tipo desejado. Para cada linha, o método tenta criar um novo item na lista do objeto, caso ele satisfaça as restrições do banco de dados. Retorna uma mensagem dizendo quais ids foram adicionados e quais não foram (por qualquer motivo). O link para acesso é URL_SERVIDOR/OBJETO/import
- Método de importar atualizações: similar ao anterior, mas tenta atualizar itens existentes ao invés de criá-los. Nesse caso, não é necessário passar todos os campos, somente os fornecidos serão atualizados. Retorna uma mensagem dizendo quais ids foram atualizados e quais não foram (por qualquer motivo). URL_SERVIDOR/OBJETO/import_update

Além dos métodos genéricos, há quatro métodos específicos:

- Método linhas por parada: tipo GET que retorna os ids de todas as linhas que contém uma parada específica, cujo id é passado no link. O link para acesso é URL_SERVIDOR/Paradas/Linhas/id
- Método veículos por linha: tipo GET que retorna os ids de todos os veículos vinculado a uma linha específica, cujo id é passado no link. O link para acesso é URL_SERVIDOR/Linhas/Veiculos/id
- Método remove paradas de linha: tipo POST que tenta remover certas paradas de uma linha, cujo id é passado no link. As paradas a remover devem ser passadas no corpo do POST e parametrizadas com 'Paradas'. Todas as paradas informadas serão desvinculadas da linha. Se algum parada não era previamente vinculada, ela é ignorada. Retorna um erro caso a linha não exista ou se não foram removidas paradas (por ids inválidos). O link para acesso é URL_SERVIDOR/Linhas/removestops/id
- Método paradas próximas: tipo GET que recebe parâmetros VALOR_LAT e VALOR_LONG referentes à latitude e longitude de um local e retorna uma lista com as paradas mais próximas desse ponto. A ordem de preferência dos resultados é: as 10 paradas mais próximas com distância menor que 3km ou uma parada mais próxima com distância maior que 3km. O link para acesso é URL_SERVIDOR/Paradas/near?lat=VALOR_LAT&long=VALOR