

Distance Speed API Documentation

File Name = `api_distance_speed.py`.

<https://depot-monitoring.chartr.in/>

API Host

How To Serve Flask Applications with Gunicorn and Nginx on Ubuntu 22.04 | DigitalOcean

In this guide, you will build a Python application using the Flask microframework on Ubuntu 22.04. The majority of this tutorial is about how to set up the ...

<https://www.digitalocean.com/community/tutorials/how-to-serve-flask-applications-with-gunicorn-and-nginx-on-ubuntu-22-04>



API Hosted on Server using Gunicorn

- API Call Example URL = https://depot-monitoring.chartr.in/overspeeding_data/2023-06-14
In Example, date = 2023-06-14
- Use Date Format = YYYY-MM-DD (iso format)

Time Filter

- Use Timestamp Format = YYYY-MM-DDTHH:MM:SS (iso format)
Example Timestamp = 2023-05-18T12:43:52
- Timestamp2 must be greater than timestamp1

Speed Filter

- Displays table entries with speed values greater than or equal to the value provided by user.

Bus Number Filter

- Displays table entries for a single bus specified by the user

Bus List Filter

- If the user wants to extract records of a list of specific buses, use this filter.
- Send a POST request to the API with the list of buses in the JSON format in the request body.
- See example below in API Examples

Get Overspeeding Instances

```
@app.route('/overspeeding_data/<date>')
@app.route('/overspeeding_data/<date>/time_filter/<timestamp1>')
@app.route('/overspeeding_data/<date>/time_filter/<timestamp1>/<timestamp2>')

# SPEED FILTER
@app.route('/overspeeding_data/<date>/<int:speed>')

# BUS FILTER
@app.route('/overspeeding_data/<date>/<bus_num>')
@app.route('/overspeeding_data/<date>/<bus_num>/<int:speed>')
@app.route('/overspeeding_data/<date>/<bus_num>/time_filter/<timestamp1>')
@app.route('/overspeeding_data/<date>/<bus_num>/time_filter/<timestamp1>/<timestamp2>')

# DEPOT FILTER
@app.route('/overspeeding_data/<date>/depot/<depot_name>')
@app.route('/overspeeding_data/<date>/<int:speed>/depot/<depot_name>')
@app.route('/overspeeding_data/<date>/depot/<depot_name>/time_filter/<timestamp1>')
@app.route('/overspeeding_data/<date>/depot/<depot_name>/time_filter/<timestamp1>/<timestamp2>')

# BUS LIST FILTER
@app.route('/<overspeeding_data>/<date>/bus_list', methods=["POST"])
@app.route('/<overspeeding_data>/<date>/bus_list/<int:speed>', methods=["POST"])
@app.route('/<overspeeding_data>/<date>/bus_list/time_filter/<timestamp1>', methods=["POST"])
@app.route('/<overspeeding_data>/<date>/bus_list/time_filter/<timestamp1>/<timestamp2>', methods=["POST"])
```

Get Distance Data

```
@app.route('/distance_data/<date>')
@app.route('/distance_data/<date>/time_filter/<timestamp1>')
@app.route('/distance_data/<date>/time_filter/<timestamp1>/<timestamp2>')

# BUS NUMBER FILTER
@app.route('/distance_data/<date>/<bus_num>')
@app.route('/distance_data/<date>/<bus_num>/time_filter/<timestamp1>')
@app.route('/distance_data/<date>/<bus_num>/time_filter/<timestamp1>/<timestamp2>')

# DEPOT FILTER
@app.route('/distance_data/<date>/depot/<depot_name>')
@app.route('/distance_data/<date>/depot/<depot_name>/time_filter/<timestamp1>')
@app.route('/distance_data/<date>/depot/<depot_name>/time_filter/<timestamp1>/<timestamp2>')

# BUS LIST FILTER
@app.route('/<distance_data>/<date>/bus_list', methods=["POST"])
@app.route('/<distance_data>/<date>/bus_list/time_filter/<timestamp1>', methods=["POST"])
@app.route('/<distance_data>/<date>/bus_list/time_filter/<timestamp1>/<timestamp2>', methods=["POST"])
```

Get Speed Data

```
@app.route('/speed_data/<date>')
@app.route('/speed_data/<date>/time_filter/<timestamp1>')
@app.route('/speed_data/<date>/time_filter/<timestamp1>/<timestamp2>')

# BUS NUMBER FILTER
@app.route('/speed_data/<date>/<bus_num>')
@app.route('/speed_data/<date>/<bus_num>/time_filter/<timestamp1>')
@app.route('/speed_data/<date>/<bus_num>/time_filter/<timestamp1>/<timestamp2>')

# DEPOT FILTER
@app.route('/speed_data/<date>/depot/<depot_name>/')
@app.route('/speed_data/<date>/depot/<depot_name>/time_filter/<timestamp1>')
@app.route('/speed_data/<date>/depot/<depot_name>/time_filter/<timestamp1>/<timestamp2>')

# BUS LIST FILTER
@app.route('/<speed_data>/<date>/bus_list', methods=["POST"])
@app.route('/<speed_data>/<date>/bus_list/<int:speed>', methods=["POST"])
@app.route('/<speed_data>/<date>/bus_list/time_filter/<timestamp1>', methods=["POST"])
@app.route('/<speed_data>/<date>/bus_list/time_filter/<timestamp1>/<timestamp2>', methods=["POST"])
```

Get Fleet Utilisation

This returns the values for :

- Fleet Held
- Scheduled Fleet
- Actual Fleet on Road
- Fleet Utilisation

Fleet Utilisation = Actual Fleet on Road / Fleet Held

```
# FLEET UTILISATION
@app.route('/fleet_utilisation/<date>')
@app.route('/fleet_utilisation/<date>/depot/<depot_name>')

# Correct set of values for agency = {'dct', 'dimts'}
@app.route('/fleet_utilisation/<date>/agency/<agency>')

# Correct set of values for bus_type = {'ac', 'nac', 'electric'}
@app.route('/fleet_utilisation/<date>/bus_type/<bus_type>')
```

Get Duty Efficiency

Duty Efficiency = (Number of Actual Duties == Scheduled Duties) / Number of Scheduled Duties

```
# DUTY EFFICIENCY
@app.route('/duty_efficiency/<date>')
@app.route('/duty_efficiency/<date>/<bus_num>')
@app.route('/duty_efficiency/<date>/depot/<depot_name>')

# Correct set of values for agency = {'dct', 'dimts'}
@app.route('/duty_efficiency/<date>/agency/<agency>')

# Correct set of values for bus_type = {'ac', 'nac', 'electric'}
@app.route('/duty_efficiency/<date>/bus_type/<bus_type>')
```

Schedule Adherence

@Ayush Madan can you please check these once

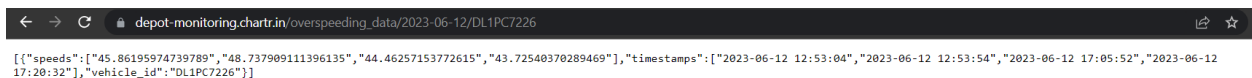
```
# SCHEDULE ADHERENCE
# - This function returns schedule adherence - scheduled duty start time vs actual outshed
@app.route('/schedule_adherence/<date>')
@app.route('/schedule_adherence/<date>/<bus_num>')
@app.route('/schedule_adherence/<date>/depot/<depot_name>')

# Correct set of values for agency = {'dct', 'dimts'}
@app.route('/schedule_adherence/<date>/agency/<agency>')

# Correct set of values for bus_type = {'ac', 'nac', 'electric'}
@app.route('/schedule_adherence/<date>/bus_type/<bus_type>')
```

Example API Call Result

1. This API call shows overspeeding instances of DL1PC7226 on 12th June 2023.
 - a. speed[i] is recorded at timestamp[i]



2. Below Image shows overspeeding instances of DL1PC0797 at in the time range :
2023-06-14 12:53:12 ≤ time ≤ 2023-06-14 17:16:13

4. Fleet Utilisation

Below are the results for fleet utilisation of “gtk” depot on 11th July 2023

```
← → ↻ 🏠 🔒 depot-monitoring.chartr.in/fleet_utilisation/2023-07-11/depot/gtk  
{ "depot": "gtk", "fleet_held": 113, "fleet_on_road": 104, "fleet_scheduled": 104, "fleet_utilisation": 0.9203539823008849 }
```

5. Duty Efficiency

Below is the duty efficiency figure for electric buses on 11th July 2023

```
← → ↻ 🏠 🔒 depot-monitoring.chartr.in/duty_efficiency/2023-07-11/bus_type/electric  
{"bus_type": "electric", "duty_efficiency": 0.8276143790849674 }
```

Restart service

```
# check status  
sudo systemctl status api_distance_speed.service  
  
# restart after changes  
sudo systemctl restart api_distance_speed.service
```