

# 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](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 = {'dtc', '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

```
# 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>')
```

## Scheduled Distance and Actual Distance

This tells what distance is scheduled for a bus for the day and how much distance it has actually covered in realtime.

Both scheduled distance and actual distance of a bus keep on updating throughout the day.

This service is not be available during 12 am to 3 am.

```
# SCHEDULED AND ACTUAL DISTANCE
# - This function returns scheduled distance and actual distance covered for a bus on a day
@app.route('/scheduled_and_actual_dist/<date>/<bus_num>')
```

## Average Distance Covered by a Bus in a Date Range

This tells how much average distance per day covered by a bus between two dates.

- Always ensure `start_date ≤ end_date`
- The “not\_found\_dates” field in API response gives a list of the dates when data for the particular bus was not found.
- See example usage below for more clarity on this field

```
# AVERAGE DISTANCE COVERED
# - This function return average distance (km) covered by a bus between two date values
@app.route('/calculate_average_distance/<start_date>/<end_date>/<bus_num>')
```

## Average Duty Efficiency in a Date Range

This returns average duty efficiency between two dates.

Always ensure `start_date ≤ end_date`

The “not\_found\_dates” field in API response gives a list of the dates when data for the particular bus was not found.

See example usage below for more clarity on this field

```
# AVERAGE DUTY EFFICIENCY
# - This function return average duty efficiency between two date values
@app.route('/calculate_average_duty_efficiency/<start_date>/<end_date>/')

@app.route('/calculate_average_duty_efficiency/<start_date>/<end_date>/<bus_num>')
@app.route('/calculate_average_duty_efficiency/<start_date>/<end_date>/depot/<depot_name>')

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

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

## Average Fleet Utilisation in a Date Range

- This returns average fleet utilisation between two dates.
- Always ensure `start_date ≤ end_date`
- The “not\_found\_dates” field in API response gives a list of the dates when data for the particular bus was not found.
- See example usage below for more clarity on this field

```
# AVERAGE FLEET UTILISATION
# - This function return average fleet utilisation between two date values
@app.route('/calculate_average_fleet_utilisation/<start_date>/<end_date>/')

@app.route('/calculate_average_fleet_utilisation/<start_date>/<end_date>/depot/<depot_name>')

# Correct set of values for agency = {'dttc', 'dimts'}
@app.route('/calculate_average_fleet_utilisation/<start_date>/<end_date>/agency/<agency>')

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

## Total Distance Covered

- This returns total distance covered by a bus between two date values.
- Always ensure `start_date ≤ end_date`
- The “not\_found\_dates” field in API response gives a list of the dates when data for the particular bus was not found.
- **For a long date range or high number of buses, the endpoint may take some time to give response**

```
# TOTAL DISTANCE COVERED
# - This function returns total distance covered by a bus between two date values
@app.route('/calculate_total_distance/<start_date>/<end_date>/')

@app.route('/calculate_total_distance/<start_date>/<end_date>/bus_num/')
@app.route('/calculate_total_distance/<start_date>/<end_date>/depot/<depot_name>')

# Correct set of values for agency = {'dttc', 'dimts'}
@app.route('/calculate_total_distance/<start_date>/<end_date>/agency/<agency>')

# Correct set of values for bus_type = {'ac', 'nac', 'electric'}
@app.route('/calculate_total_distance/<start_date>/<end_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]`



```
[{"speeds":["45.86195974739789","48.737909111396135","44.46257153772615","43.72540370289469"],"timestamps":["2023-06-12 12:53:04","2023-06-12 12:53:54","2023-06-12 17:05:52","2023-06-12 17:20:32"],"vehicle_id":"DL1PC7226"}]
```


## 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

```
← → ↻ depot-monitoring.chartr.in/speed_data/2023-06-14/DL1PC0797/time_filter/2023-06-14T12:53:12/2023-06-14T17:16:13
[{"speed_kmph":
["16.14547764434234","1.4432180997670572","8.274976246498607","21.45241466240287","28.09116743401433","26.760869532202705","18.30652872996213","11.626943252724182","16.940268604246445","10.19393820653134","33.695
12881954091","38.783458955453646","0","0","11.963067506519083","8.89532515018457","4.514269978524674","37.15802697327852","26.44855009271956","9.974900953397626","1.9027285510823375","0","7.836477718582801","0.15
984546422161336","0.8008584845246913","0","0","0","0","0","0","0.5948094700067881","0","0","0","0.4105712327745892","0.07400782122083982","0","0","1.10933705700212","0","0","0.5434953023207021","0","0","0
","0","0","0","0","2.0741810931306608","0","0","17.60389904241185","0","28.770401200641555","0","0","21.514042128110775","24.060588747656","0","3.8516889023913956","3.745659984205207","0","3.06659802018385","8.59
7704444196473","19.17806108973326","17.20176488638533","0","0","0","18.43786084772337","18.16801624801724","27.754519052754645","27.472963029182488","12.30403518528589","40.18166057785259","10.14603429694558
3","35.62567154023491","27.6708590982306","1.1800715681371918","28.16600017486117","17.862403818974975","23.991006611588727","22.228952864985345","26.112811299754924","10.42294358647985","19.18579645411912","30
.05794243497238","19.725449310023713","17.388966997086037","28.08045229450659","15.850095445959113","9.660426676000954","26.987279095551262","21.024188582424664","14.682517327888633","15.083290941512839","2.99631
70163759276","14.104030511486949","21.335660231015527","16.490078091579264","30.41902581639874","12.26180040851389","5.4073198003864835","15.837859826163363","25.16425999670873","18.691289094132387","10.180691169
129732","21.241441471400492","14.209417461632437","21.106801168178684","26.30002454572581","19.9287126508768","14.711752404874757","29.01465084742998","3.6906037358562807","24.053864630590024","25.6682701117528",
"14.96393716281297","28.80872347191305","48.997914945097335"],"timestamps":["2023-06-14 12:53:52","2023-06-14 12:54:32","2023-06-14 13:00:52","2023-06-14 13:02:22","2023-06-14 13:03:42","2023-06-14
13:06:52","2023-06-14 13:09:22","2023-06-14 13:11:32","2023-06-14 13:12:42","2023-06-14 13:15:22","2023-06-14 13:16:32","2023-06-14 13:19:02","2023-06-14 13:19:02","2023-06-14
13:25:42","2023-06-14 13:26:42","2023-06-14 13:28:12","2023-06-14 13:30:02","2023-06-14 13:33:52","2023-06-14 13:37:05","2023-06-14 13:38:05","2023-06-14 13:38:15","2023-06-14 13:39:05","2023-06-14
13:45:00","2023-06-14 13:47:35","2023-06-14 13:49:35","2023-06-14 13:51:45","2023-06-14 13:53:53","2023-06-14 13:55:53","2023-06-14 13:58:03","2023-06-14 14:00:53","2023-06-14 14:03:03","2023-06-14
14:05:13","2023-06-14 14:07:13","2023-06-14 14:09:23","2023-06-14 14:11:13","2023-06-14 14:13:33","2023-06-14 14:15:43","2023-06-14 14:17:53","2023-06-14 14:18:33","2023-06-14 14:18:33","2023-06-14
14:18:33","2023-06-14 14:18:33","2023-06-14 14:28:18","2023-06-14 14:29:18","2023-06-14 14:29:58","2023-06-14 14:31:16","2023-06-14 14:31:16","2023-06-14 14:35:46","2023-06-14 14:37:56","2023-06-14
14:40:06","2023-06-14 14:40:06","2023-06-14 14:40:56","2023-06-14 14:40:56","2023-06-14 14:40:56","2023-06-14 14:47:16","2023-06-14 14:47:16","2023-06-14 14:47:36","2023-06-14 14:47:36","2023-06-14
14:47:36","2023-06-14 14:49:06","2023-06-14 14:52:06","2023-06-14 14:52:06","2023-06-14 14:54:16","2023-06-14 14:56:36","2023-06-14 14:56:36","2023-06-14 14:59:26","2023-06-14 15:06:56","2023-06-14
15:18:16","2023-06-14 15:20:46","2023-06-14 15:20:46","2023-06-14 15:20:46","2023-06-14 15:20:46","2023-06-14 15:20:46","2023-06-14 15:32:51","2023-06-14 15:34:51","2023-06-14 15:37:01","2023-06-14
15:39:11","2023-06-14 15:41:11","2023-06-14 15:43:11","2023-06-14 15:45:21","2023-06-14 15:47:11","2023-06-14 15:49:21","2023-06-14 15:51:31","2023-06-14 15:53:27","2023-06-14 15:55:43","2023-06-14
15:57:43","2023-06-14 15:59:43","2023-06-14 16:01:53","2023-06-14 16:03:53","2023-06-14 16:06:03","2023-06-14 16:08:03","2023-06-14 16:10:13","2023-06-14 16:12:13","2023-06-14 16:14:13","2023-06-14
16:16:23","2023-06-14 16:18:33","2023-06-14 16:20:33","2023-06-14 16:22:43","2023-06-14 16:24:43","2023-06-14 16:26:53","2023-06-14 16:28:43","2023-06-14 16:31:03","2023-06-14 16:33:03","2023-06-14
16:35:03","2023-06-14 16:37:13","2023-06-14 16:39:13","2023-06-14 16:41:13","2023-06-14 16:43:13","2023-06-14 16:45:13","2023-06-14 16:47:23","2023-06-14 16:49:23","2023-06-14 16:51:23","2023-06-14
16:53:23","2023-06-14 16:55:33","2023-06-14 16:57:33","2023-06-14 16:59:33","2023-06-14 17:01:43","2023-06-14 17:03:43","2023-06-14 17:05:53","2023-06-14 17:07:53","2023-06-14 17:10:03","2023-06-14
17:12:03","2023-06-14 17:14:13","2023-06-14 17:16:13"],"vehicle_id":"DL1PC0797"]}]
```

## 3. Bus List Filter

- Send a POST request to the API with the list of buses in the JSON format in the request body
- This request fetched speed data on 25th June, 2023 for list of buses given below in the time range :
- 2023-06-25 12:43:52 ≤ time ≤ 2023-06-25 14:43:52

 [https://depot-monitoring.chartr.in/overspeeding\\_data/2023-06-25/bus\\_list/time\\_filter/2023-06-25T12:43:52/2023-06-25T14:43:52](https://depot-monitoring.chartr.in/overspeeding_data/2023-06-25/bus_list/time_filter/2023-06-25T12:43:52/2023-06-25T14:43:52)

POST

https://depot-monitoring.chartr.in/overspeeding\_data/2023-06-25/bus\_list/time\_filter/2023-06-25T12:43:52/2023-06-25T14:43:52

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

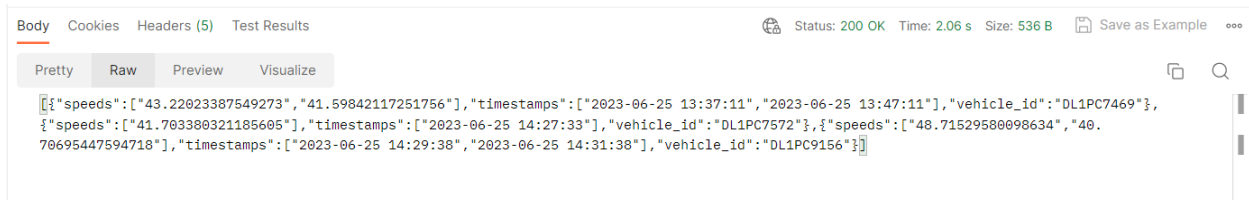
binary

GraphQL

JSON

```
1  {
2    "bus_list": ["DL1PC7469", "DL1PC9156", "DL1PC9740", "DL1PC7572", "DL1PC9579"]
3  }
```

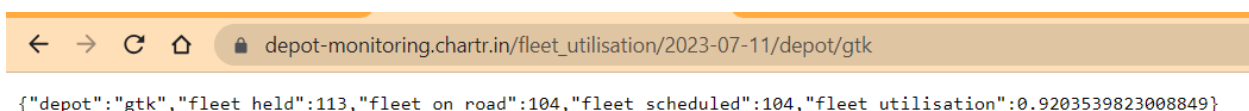
Input



Output

#### 4. Fleet Utilisation

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



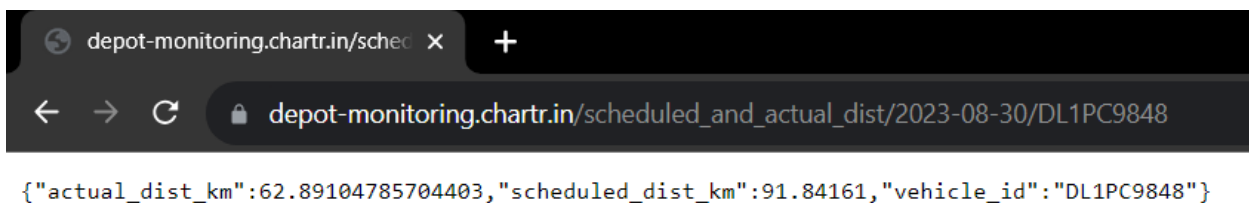
#### 5. Duty Efficiency

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



#### 6. Scheduled and Actual Distance

Below is scheduled and actual distance values for bus number = DL1PC9848 on 30th August 2023



#### 7. Average Distance Covered by a Bus in a Date Range

Below is average distance per day covered by bus number DL1PC9484 from 19th August 2023 to 25th August 2023.

Data is not present for dates ["2023-08-19", "2023-08-20", "2023-08-21"]. Hence this average distance takes



into account only 4 days, i.e from 22nd August 2023 to 25th August 2023.

Average distance = Sum of Total Distances covered in dates from 22nd August 2023 to 25th August 2023  
/ Number of Days (i.e 4)

```
depot-monitoring.chartr.in/calculate_average_distance/2023-08-19/2023-08-25/DL1PC9484
{"average_distance_covered_km":145.8462750424903,"end_date":"2023-08-25","not_found_dates":["2023-08-19","2023-08-20","2023-08-21"],"start_date":"2023-08-19","vehicle_id":"DL1PC9484"}
```

Below is average distance per day covered by bus number DL1PC9484 from 27th August 2023 to 1st September 2023.

Data for the given bus was present for all the dates lying in date range, hence “not\_found\_dates” field shows empty list.

```
depot-monitoring.chartr.in/calculate_average_distance/2023-08-27/2023-09-01/DL1PC9484
{"average_distance_covered_km":162.30179005488935,"end_date":"2023-09-01","not_found_dates":[],"start_date":"2023-08-27","vehicle_id":"DL1PC9484"}
```

## 8. Average Duty Efficiency in a Date Range

Below is Average Duty Efficiency for Depot = “and” from 7th October 2023 to 13th October 2023

```
depot-monitoring.chartr.in/calculate_average_duty_efficiency/2023-10-07/2023-10-13/depot/and
{"average_duty_efficiency":"83.48%","depot":"and","end_date":"2023-10-13","not_found_dates":[],"start_date":"2023-10-07"}
```

## 9. Average Fleet Utilisation in a Date Range

Below is Average Fleet Utilisation for depot = “gtk” from 22nd November 2023 to 29th November 2023

```
depot-monitoring.chartr.in/calculate_average_fleet_utilisation/2023-11-22/2023-11-29/depot/gtk
{"average_fleet_utilisation":"79.96%","depot":"gtk","end_date":"2023-11-29","not_found_dates":[],"start_date":"2023-11-22"}
```

## 9. Total Distance Covered

Below is total distance covered by bus type = “ac” from 7th November 2023 to 10th November 2023

```
← → ↻ 📁 depot-monitoring.charttr.in/calculate_total_distance/2023-11-07/2023-11-10/bus_type/ac
[{"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 634.66349511063, "vehicle_id": "DL1PC7002"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 717.4083620524694, "vehicle_id": "DL1PC7003"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 128.11107774345692, "vehicle_id": "DL1PC7004"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 630.8259430042776, "vehicle_id": "DL1PC7008"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 319.53222923357606, "vehicle_id": "DL1PC7010"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 472.4870501903092, "vehicle_id": "DL1PC7011"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 665.5235924731326, "vehicle_id": "DL1PC7012"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 726.733179479399, "vehicle_id": "DL1PC7013"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 666.73705160251478, "vehicle_id": "DL1PC7014"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 740.5387803638178, "vehicle_id": "DL1PC7018"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 472.4870501903092, "vehicle_id": "DL1PC7018"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 659.4513356317701, "vehicle_id": "DL1PC7092"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 638.1357219625022, "vehicle_id": "DL1PC7094"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 528.1426819396283, "vehicle_id": "DL1PC7135"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 661.5286021588472, "vehicle_id": "DL1PC7136"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 692.0945534405262, "vehicle_id": "DL1PC7137"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 588.352045546916, "vehicle_id": "DL1PC7139"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 687.171556162155, "vehicle_id": "DL1PC7142"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 558.3085354616239, "vehicle_id": "DL1PC7147"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 422.948894992305, "vehicle_id": "DL1PC7150"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 710.4034184571226, "vehicle_id": "DL1PC7153"}, {"bus_type": "ac", "end_date": "2023-11-10", "not_found_dates": [], "start_date": "2023-11-07", "total_distance_covered_km": 710.4034184571226, "vehicle_id": "DL1PC7153"}]
```

## Restart service

```
# check status
sudo systemctl status api_distance_speed.service

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

## View Logs

```
journalctl -u api_distance_speed.service

# View real time logs
journalctl -fu api_distance_speed.service

# filter logs by time
journalctl -u api_distance_speed.service --since "30 minutes ago"

# search for errors
journalctl -u api_distance_speed.service | grep "error"
```