



GeoTrackers RESTful API

COMPANY PROPRIETARY
CONFIDENTIAL

Revision: 0.2
Jan 3, 2016

REVISION HISTORY

Revision	Description	Date	Author
0.1	Initial Version	Aug 5, 2015	AKM
0.2	Updates for TAG management	Jan 3, 2016	AP

TABLE OF CONTENTS

1 Introduction.....	2
2 High Level Architecture.....	2
2.1 M2M Interface – RESTful API.....	4
2.1.1 Current Location of all the User Vehicles.....	4
2.1.2 Calculate Distance moved by a Vehicle.....	5
2.1.3 Vehicle Daily Report.....	6
2.1.4 Vehicle List assigned to the User.....	8
2.1.5 Vehicle movement history as TripLegs (durations of halts and movements).....	9
2.1.6 Vehicle movement history as Locations.....	11
2.1.7 Vehicle Location on or before a specific timestamp.....	12
2.1.8 Vehicle Current Location.....	13
2.1.9 Current Location of list of Selected Vehicles.....	14
2.1.10 Create TAG for vehicle.....	15
2.1.11 Delete TAG for vehicle.....	16
2.1.12 Update TAG for vehicle.....	17
2.1.13 Create TAGs for all the Vehicles of a User.....	18

1 Introduction

Geotrackers Automatic Vehicle/Personnel Location System provides location and GIS services to logistics and Field Force Management (FFM) applications. The solution provides the following features:

- GPS location data capture engine. The engine can accept data from GPRS/GSM/CDMA/3G based devices.
- OpenGIS compliant map server to display vehicles as an overlay over the maps. The map server can read data in various GIS formats such as OSM, shape files, PostGIS, Mapinfo etc.
- High performance map engine to serve maps to the large number of web based clients
- Web based clients, Android and iOS apps to access data practically from anywhere over any type of connection
- Various GIS features such as geofencing, vicinity alerts etc
- Accurate KM logs
- Various MIS reports such as trip history, trip playback, daily summary, fleet summary, site visit reports etc.
- Webservices based API for external ERPs to interface and retrieve data
- Pluggable database engine which supports any J2EE and OpenGIS compliant database engine.

The solution uses highly scalable distributed J2EE based architecture. The architecture allows tracking of thousands of units from the same installation.

2 High Level Architecture

The following diagram depicts high-level architecture of the system. Each component and sub component of the system can run on the same or different hardware- making the architecture highly scalable with built-in redundancy. The blocks marked with blue shade are company proprietary.

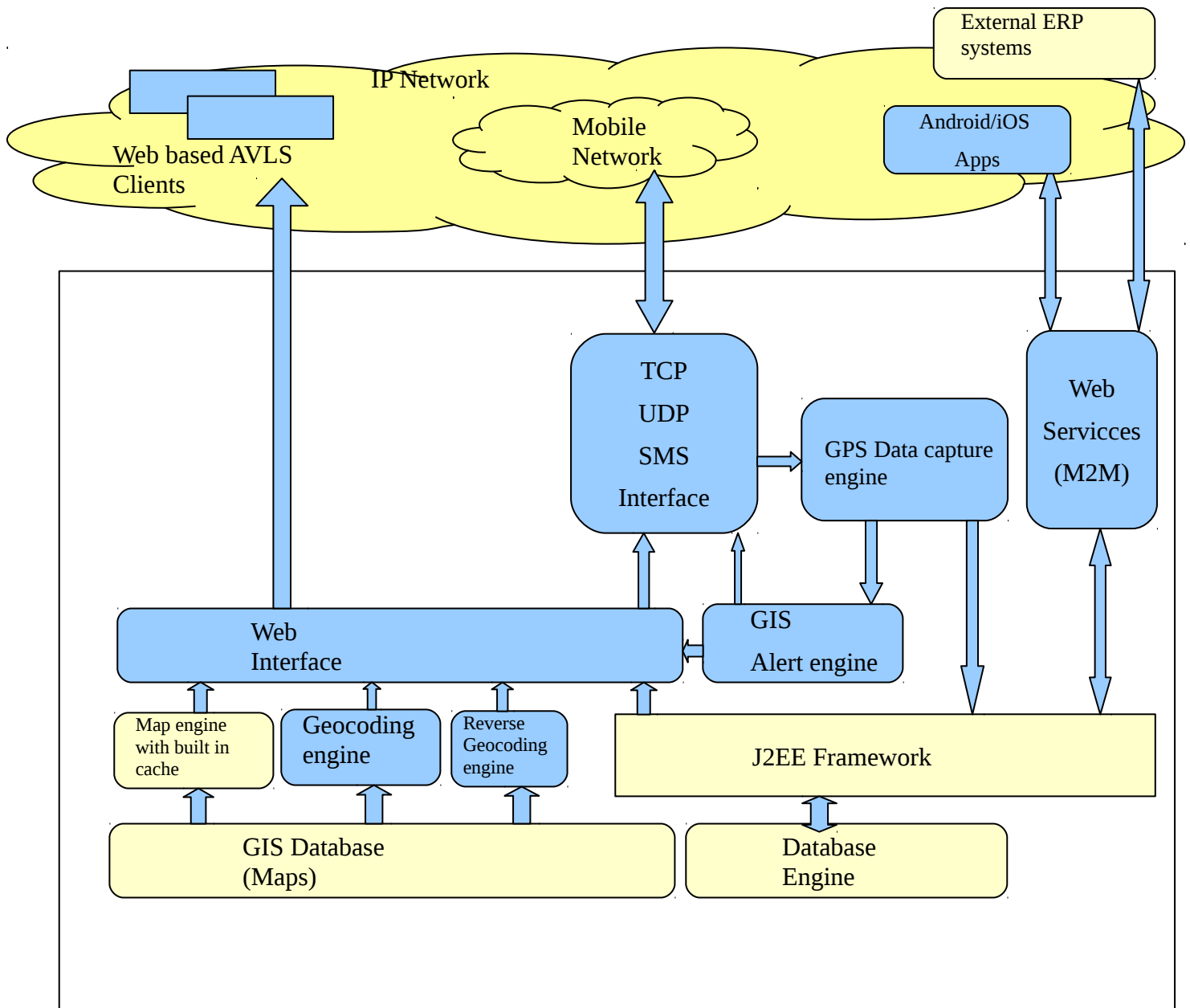


Figure 1: Block Level Architecture of the GeoAVL

2.1 M2M Interface – RESTful API

Geotrackers M2M interface provides API to interface with external ERP systems as well as consumption from Geotrackers' own Android and iOS apps. The API uses basic authentication.

The time parameter in all the API URLs can be given as EPOCH time (milliseconds from 1 Jan 1970 00:00:00 GMT) or in `yyyyMMddHHmmss` format in IST.

The URL should be encoded using URL encoding as per rfc3986.

2.1.1 Current Location of all the User Vehicles

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getUsrLatestLocation>

Params:

None. The system takes the user name internally from the logged in user name.

Purpose:

To get latest locations and analog data (temperature etc.) of all the vehicles assigned to the user.

Example:

When the API is called, the result will have an array of vehicles locations and analog data - latitude, longitude, speed, status, temperature, fuel etc. The timestamp (EPOCH time in milliseconds from 1 Jan 1970 00:00:00 GMT) gives the date and time for which data was last updated for that vehicle.

Sample Output for a halted vehicle

```
{"CUST-7112-HR 55N":  
[{"systime":1459316011363,"exceptionBM":1,"virtualName":"","speed":0.0,"direction":180,"haltedSince":"1  
days 3 hrs 58 mins ","elevation":198.0,"timestamp":1459316009000,"distance":0.0,"locStr":"MS FROZEN  
OFFICE, Delhi, DL","noDataSince":"9 sec","latitude":28.675093,"analogData":"Temperature 30.86  
Â°C","movingSince":"0 sec","longitude":77.142365,"regNo":"CUST-8512-HR 55N","bmStr":"Long Halt"}]}
```

Sample Output for a moving vehicles

```
"CUST-DL 3CV 1683":  
[{"systime":1435934071758,"exceptionBM":0,"virtualName":"","speed":11.0,"direction":177,"haltedSince":"0  
sec","elevation":232.0,"timestamp":1435934044000,"distance":0.102830835,"locStr":"0.4 km from  
Vishwakarma Chowk, Gurgaon, HR","noDataSince":"1 mins  
","latitude":28.525166,"analogData":"","movingSince":"43 mins ","longitude":77.10171,"regNo":"CUST-DL  
3CV 1683","bmStr":""}]
```

2.1.2 Calculate Distance moved by a Vehicle

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getTotalDistance/{vhRegNo}/{btime}/{etime}>

Params:

vhRegNo: Vehicle registration number exactly as registered by Geotrackers'

btime: begin time

etime: etime

Purpose:

The API will reply with distance Kms moved by the vehicle between btime and etime

Example:

If the above API is called using the following URL

Dates given as EPOC time

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getTotalDistance/CUST-DL%201LT%204476/1436170786457/1436174386457>

or

Dates given in yyyyMMddHHmmss format (IST)

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getTotalDistance/CUST-DL%201LT%204476/20150706134946/20150706144946>

The response is

```
{"totDistKm":4.3429813}
```

2.1.3 Vehicle Daily Report

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getDailyReport/{vhRegNo}/{btime}/{etime}/{stopSecs}/{defaultHaltSec}/{speedLimit}>

Params:

vhRegNo: Vehicle registration number exactly as registered by Geotrackers'

btime: begin time

etime: etime

stopSecs: Halts longer than these seconds will be considered as "Long Halt" and will be reported

defaultHaltSec: Halts less than this will be included as movement and not halted

speedLimit: Speed (KM/h) higher than this will be considered overspeeding

Note: Maximum 3 days data allowed in one call

Purpose:

The API will reply with daily movement, halt and exception summary

Example:

Get daily report of CUST-DL 1LT 4476, from 06/Jul/2015 00:00 IST to 07/Jul/2015 00:00 IST. 180 seconds are both halt limits and 50 kmph is speed limit

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getDailyReport/CUST-DL%201LT%204476/1436034600000/1436121000000/180/180/50>

The reply is as following

```
{"CUST-DL 1LT 4476":[{"endTsStr":"06\07\2015 00:00:00
IST","kms":0.0,"haltDurationSec":86321,"startLocStr":"0.7 km from Okhla Phase - I, Delhi,
DL","endTs":1436121000000,"movementDurationSec":0,"avgSpeed":0.0,"beginTsStr":"05\07\2015 00:00:00
IST","maxSpeed":0.0,"beginTs":1436034600000,"endLocStr":"0.7 km from Okhla Phase - I, Delhi, DL","regNo":"CUST-
DL 1LT 4476","NDDurationSec":0}]}
```

"regNo": the vehicle

"kms":total movement in kms

"beginTsStr":End date in human readable format

"endTsStr": Begin date in human readable format

"haltDurationSec":number of seconds vehicle was halted

"movementDurationSec": total number of seconds for which vehicle was moving

"NDDurationSec": total number of seconds in which there was no data (data blackout)

"startLocStr": human readable begin location

"endLocStr": human readable end location

"beginTs": begin timestamp

"endTs":end timestamp

"avgSpeed":average speed in kmph

"maxSpeed": max speed in kmph

2.1.4 Vehicle List assigned to the User

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getUsrVehicles>

Params:

none

Purpose:

The API will reply with all the vehicles assigned to the user

Example:

When the above URL is called - an array with the list of vehicle registration number is returned

```
{"CUST":["CUST- HR 55K 6653","CUST-DL 1LE 9637","CUST-DL 1LG 5851","CUST-DL 1LG 6158","CUST-DL 1LG 6778","CUST-DL 1LG 7635","CUST-DL 1LG 8681","CUST-DL 1LG 8753","CUST-DL 1LH 7245","CUST-DL 1LJ 8423","CUST-DL 1LK 0935","CUST-DL 1LK 1090","CUST-DL 1LK 1866","CUST-DL 1LK 4394","CUST-DL 1LK 5359","CUST-DL 1LK 7440"]}
```

2.1.5 Vehicle movement history as TripLegs (durations of halts and movements)

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getHistoricalLocations>

`/ {vhRegNo} / {speedLimit} / {btime} / {etime}`

Params:

Same as above APIs

Purpose:

The API will reply with tripslegs (consecutive movement and halts)

Example:

When the above URL is called with timestamps of 06/Jul/2015 00:00 IST to 07/Jul/2015 00:00 IST and 50 kmph as speed limit

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getHistoricalLocations/CUST-DL%201LT%204476/50/1436034600000/1436121000000>

The response is in terms of TripLegs (halts and movements)

```
{ "CUST-DL 1LT 4476": { "tripLegArrList": [ { "type": 9, "startTime": 1436034600000, "endTime": 1436109184000, "distKM": 0.0, "speedKmph": 0.0, "maxSpeedKmph": 0.0, "status": "Long Halt, No Data", "startLocStr": "M416, Pune, MH", "endLocStr": "", "duration": "20 h 43 m", "tlTotalDist": 0.0, "tlTotalDurationStr": "20 h 43 m", "subLegs": [ { "exBM": 1, "latLng": [18.4958, 73.7991] } ] }, { "type": 2, "startTime": 1436109204000, "endTime": 1436114060000, "distKM": 22.49631, "speedKmph": 36.447666, "maxSpeedKmph": 85.229, "status": "Overspeeding", "startLocStr": "0.1 km from M416, Pune, MH", "endLocStr": "0.3 km from Tata Motors, Pune, MH", "duration": "37 m", "tlTotalDist": 22.49631, "tlTotalDurationStr": "21 h 20 m", "subLegs": [ { "exBM": 0, "latLng": [18.4966, 73.8001], [18.4965, 73.8008], [18.4963, 73.8015], [18.4968, 73.802], [18.4986, 73.8021], [18.5006, 73.8022], [18.5021, 73.8023], [18.5027, 73.802], [18.5035, 73.8021], [18.5043, 73.8023], [18.5054, 73.8025], [18.5064, 73.8027], [18.5068, 73.8027], [18.5073, 73.801], [18.5075, 73.7995], [18.5077, 73.798], [18.5079, 73.7965], [18.5079, 73.795], [18.5081, 73.7934], [18.5082, 73.7925], [18.5084, 73.791], [18.5082, 73.7895], [18.5077, 73.7881], [18.507, 73.7875], [18.5071, 73.7858], [18.5072, 73.7853] }, { "exBM": 2, "latLng": [18.5074, 73.7829], [18.5081, 73.7798], [18.5087, 73.7766], [18.5098, 73.7736], [18.5118, 73.771], [18.5142, 73.7695], [18.5166, 73.769] ] }, { "exBM": 0, "latLng": [18.5182, 73.7686], [18.5196, 73.7688], [18.5219, 73.7695], [18.5241, 73.7712] } ], { "exBM": 2, "latLng": [18.5262, 73.7734], [18.5276, 73.7768], [18.5304, 73.779], [18.5337, 73.7802] } ], { "exBM": 0, "latLng": [18.5362, 73.7808] } ], { "exBM": 2, "latLng": [18.5385, 73.7812], [18.5409, 73.7792], [18.5429, 73.7765], [18.5447, 73.7746], [18.5476, 73.7734], [18.5506, 73.7724], [18.5541, 73.7711], [18.5573, 73.77] } ], { "exBM": 0, "latLng": [18.5594, 73.7693], [18.5618, 73.7684] } ], { "exBM": 2, "latLng": [18.5647, 73.7673], [18.5682, 73.7661], [18.5717, 73.7647], [18.5754, 73.7634], [18.5794, 73.7619], [18.5834, 73.7604], [18.5868, 73.7591], [18.5895, 73.7581] } ], { "exBM": 0, "latLng": [18.5921, 73.7571], [18.594, 73.7564], [18.5964, 73.7554], [18.5987, 73.7546], [18.6011, 73.7537], [18.603, 73.7528], [18.6048, 73.752], [18.6054, 73.752], [18.6061, 73.753], [18.6066, 73.7545], [18.607, 73.7555], [18.6079, 73.7566], [18.6096, 73.7584], [18.6108, 73.7601], [18.6117, 73.7612], [18.613, 73.7628], [18.6139, 73.7638], [18.615, 73.7649], [18.6152, 73.7651], [18.6154, 73.7654], [18.6166, 73.7666], [18.618, 73.7682], [18.6197, 73.77], [18.6208, 73.7713], [18.6217, 73.7724], [18.6231, 73.774], [18.6241, 73.7753], [18.6254, 73.7758], [18.6276, 73.7754], [18.629, 73.7752], [18.6296, 73.776], [18.6307, 73.7772], [18.6312, 73.7779], [18.6307, 73.7789], [18.6303, 73.7798], [18.6298, 73.7808], [18.6297, 73.7813], [18.6301, 73.7822], [18.6313, 73.7836], [18.6325, 73.785], [18.6332, 73.7858], [18.634, 73.7874], [18.6351, 73.7881], [18.635, 73.7882], [18.6344, 73.7879], [18.6342, 73.7884], [18.634, 73.7887] ] } ], { "type": 1, "startTime": 1436114060000, "endTime": 1436114032000, "distKM": 0.0, "speedKmph": 0.0, "maxSpeedKmph": 0.0, "status": "Long Halt, No Data", "startLocStr": "M416, Pune, MH", "endLocStr": "", "duration": "32 m", "tlTotalDist": 0.0, "tlTotalDurationStr": "32 m", "subLegs": [ { "exBM": 1, "latLng": [18.4958, 73.7991] } ] } ] } }
```

h\":0.0,\"status\\\":\\\"Long Halt\\\",\\\"startLocStr\\\":\\\"0.3 km from Tata Motors, Pune, MH\\\",\\\"endLocStr\\\":\\\"\\\",\\\"duration\\\":\\\"44 m \\\",\\\"tlTotalDist\\\":22.49631,\"tlTotalDurationStr\\\":\\\"22 h 4 m \\\",\\\"subLegs\\\":[{\\\"exBM\\\":1,\"latLng\\\":[[18.634,73.7887]]}], {\\\"type\\\":2,\"startTime\\\":1436114052000,\"endTime\\\":1436116173000,\"distKM\\\":21.037956,\"speedKmph\\\":35.374424,\"maxSpeedKmph\\\":69.6537,\"status\\\":\\\"Overspeeding\\\",\\\"startLocStr\\\":\\\"0.5 km from Tata Motors, Pune, MH\\\",\\\"endLocStr\\\":\\\"M416, Pune, MH\\\",\\\"duration\\\":\\\"35 m \\\",\\\"tlTotalDist\\\":43.534264,\"tlTotalDurationStr\\\":\\\"22 h 39 m \\\",\\\"subLegs\\\":[{\\\"exBM\\\":0,\"latLng\\\": [[18.6335,73.7867],[18.6328,73.7853],[18.6319,73.7843],[18.6306,73.783],[18.6297,73.7819],[18.6295,73.7815],[18.6297,73.7808],[18.6302,73.7795],[18.6306,73.7788],[18.6309,73.7778],[18.6302,73.7769],[18.6292,73.7758],[18.6279,73.7755],[18.6257,73.7758],[18.6243,73.7755],[18.6235,73.7746],[18.6223,73.7732],[18.6213,73.772],[18.62,73.7705],[18.6189,73.7692],[18.6174,73.7676],[18.6164,73.7664],[18.6154,73.7655],[18.6146,73.7646],[18.6132,73.7632],[18.6126,73.7625],[18.6114,73.7609],[18.6103,73.7594],[18.6092,73.7578],[18.6081,73.7567],[18.6076,73.7562],[18.6069,73.7554],[18.6064,73.7541],[18.6057,73.7526],[18.6041,73.753],[18.6021,73.7535]]}, {\\\"exBM\\\":2,\"latLng\\\":[[18.5991,73.7546],[18.5959,73.7558],[18.5932,73.7568]]},{\\\"exBM\\\":0,\"latLng\\\": [[18.5914,73.7575]]},{\\\"exBM\\\":2,\"latLng\\\":[[18.5886,73.7585],[18.5853,73.7598]]},{\\\"exBM\\\":0,\"latLng\\\": [[18.5824,73.7608]]},{\\\"exBM\\\":2,\"latLng\\\":[[18.5799,73.7618],[18.5768,73.763],[18.5743,73.7639],[18.5716,73.7649],[18.5683,73.7661],[18.5652,73.7673],[18.5623,73.7683],[18.5592,73.7694],[18.5559,73.7706],[18.5528,73.7716],[18.5497,73.7727],[18.5467,73.7738],[18.5441,73.7753],[18.5421,73.7776],[18.5401,73.7804],[18.5369,73.7812],[18.534,73.7803],[18.5308,73.7793]]},{\\\"exBM\\\":0,\"latLng\\\":[[18.5284,73.7781],[18.5277,73.7785],[18.5269,73.7797],[18.5258,73.7814],[18.5248,73.781],[18.5229,73.7802],[18.5212,73.7794],[18.5195,73.7788],[18.518,73.78],[18.517,73.7812],[18.5155,73.782],[18.5141,73.7826],[18.5121,73.7822],[18.5105,73.7821],[18.5084,73.7829],[18.5076,73.7839],[18.5075,73.7857],[18.5077,73.7876],[18.5082,73.7893],[18.5085,73.7909],[18.5083,73.7923],[18.5081,73.7939],[18.5079,73.7958],[18.5078,73.7973],[18.5077,73.7976],[18.5064,73.7976],[18.5054,73.7979],[18.5042,73.7978],[18.5033,73.7977],[18.503,73.7989],[18.5027,73.8005],[18.5023,73.8022],[18.5011,73.8023],[18.4988,73.8022],[18.497,73.8021],[18.4967,73.8021],[18.4967,73.8021],[18.4962,73.8019],[18.4965,73.8007],[18.4963,73.8],[18.496,73.7994]]}], {\\\"type\\\":1,\"startTime\\\":1436116173000,\"endTime\\\":1436120981000,\"distKM\\\":0.0,\"speedKmph\\\":0.0,\"maxSpeedKmph\\\":0.0,\"status\\\":\\\"Long Halt\\\",\\\"startLocStr\\\":\\\"M416, Pune, MH\\\",\\\"endLocStr\\\":\\\"\\\",\\\"duration\\\":\\\"1 h 20 m \\\",\\\"tlTotalDist\\\":43.534264,\"tlTotalDurationStr\\\":\\\"23 h 59 m \\\",\\\"subLegs\\\":[{\\\"exBM\\\":1,\"latLng\\\": [[18.496,73.7994]]}]]}]\"}

2.1.6 Vehicle movement history as Locations

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getPastLocations/{vhRehNo}/{btime}/{etime}/{speedlimit}/{haltlimit}>

Params:

Same as above APIs

Purpose:

The API will reply with locations

Example:

e.g. to get history from Apr 19 14:24:59 2016 IST to Apr 19 15:00:47 2016 with 50 kmph speed limit and 180 seconds halt limit

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getPastLocations/CUST55Q/1461056099000/1461058247000/50/180>

The response

```
{ "pastLocations": [ { "timestamp": 1461056160000, "speed": 0.0, "direction": 0, "latitude": 28.3787, "Temperature": 45.17, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056250000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.52, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056340000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.52, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056430000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056520000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.17, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056610000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.52, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056700000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.52, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056790000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.17, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056880000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.52, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461056970000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057060000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057150000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057239000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057330000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.52, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057420000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057509000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057599000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057689000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057779000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057869000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 46.21, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461057959000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461058049000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 46.21, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461058139000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 46.21, "longitude": 76.8342, "bmStr": "Long Halt" },
{ "timestamp": 1461058229000, "speed": 0.0, "direction": 90, "latitude": 28.3787, "Temperature": 45.86, "longitude": 76.8342, "bmStr": "Long Halt" } ] }
```

2.1.7 Vehicle Location on or before a specific timestamp

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getLocationAtTs/{vhRehNo}/{etime}>

Params:

Same as above APIs

Purpose:

The API will reply with location on or just before the specified timestamp

Example:

e.g. to get location of a vehicle on or just before Apr 19 14:26:39 2016 IST

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getLocationAtTs/CUST-R%2055Q/1461056199000>

The response

```
{"LocationAtTs":{"timestamp":1461056160000,"speed":0.0,"direction":90,"latitude":28.3787,"Temperature":45.17,"longitude":76.8342,"bmStr":"Long Halt"}}
```

Please note that the system searches data in a 30 minutes time-window.

2.1.8 Vehicle Current Location

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getLatestLocation/{vhRehNo}>

Params:

Same as above APIs

Purpose:

The API will reply with the latest location available for the vehicle

Example:

e.g. to get current location of a vehicle

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getLatestLocation/CUST-R%2055Q>

The response

```
{"location":  
{"systime":1478507243889,"exceptionBM":524288,"virtualName":"","speed":0.0,"direction":61,"haltedSince":  
"1 mins ","elevation":0.0,"timestamp":1478507242000,"distance":0.0,"locStr":"0.1 km from Erandwana, Pune,  
MH","noDataSince":"1 mins ","latitude":18.511482,"analogData":"","movingSince":"30 mins  
","longitude":73.834145,"regNo":"MadCel","bmStr":"Armed"}}
```

2.1.9 Current Location of list of Selected Vehicles

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getSelectedVehicleData>

Params:

This API expects vehicle list in HTTP POST. e.g.

```
{"vehList":{"selectedVehicle":["LenTabX","Tata Safari 7669","MadCel","GVT-918022"]}}
```

Purpose:

The API will reply with locations of all the vehicles specified in the vehList

Example:

e.g. to get location of a vehicle on or just before Apr 19 14:26:39 2016 IST

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/getSelectedVehicleData>

and HTTP POST

```
{"vehList":{"selectedVehicle":["LenTabX","Tata Safari 7669","MadCel","GVT-918022"]}}
```

2.1.10 Create TAG for vehicle

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/createTagForVehicle/{vhRegNo}/{validTilltime}?emails={emailIds}>

Params:

validTillTime: Tag expiry time. Value of 0 will imply no expiry

emails: comma separated email IDs which will be notified about TAG creation

Purpose:

Create TAG for the vehicle with an expiry date

Example:

e.g. to create TAG for a vehicle

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/createTagForVehicle/MH12GK8778/20161231120000?emails=bat@geotrackers.com>

Response:

```
{"status":"success","tag":"5606","vehicle":"Delhi_GT","expiry":1483165800000}
```


2.1.11 Delete TAG for vehicle

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/deleteTagForVehicle/{vhRegNo}/{TagNo}>

Params:

TagNo: tag to be deleted

Purpose:

Delete TAG for the vehicle

Example:

e.g. to delete a TAG for a vehicle

Response:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/deleteTagForVehicle/MH12GK8778/7123>

Response:

```
{"status": "Record deleted Successfully."}
```

2.1.12 Update TAG for vehicle

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/updateTagForVehicle/{vhRegNo}/{tag}/{expiry}?emails={emails}>

Params:

TagNo: tag to be updated with new expiry and email addresses

Purpose:

Update TAG for the vehicle

Example:

e.g. to update a TAG for a vehicle

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/updateTagForVehicle/MH12GK8778/7123/20161231120000?emails=bat@geotrackers.com>

Response:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/deleteTagForVehicle/MH12GK8778/7123>

Response:

{"status": "Record Updated Successfully."}

2.1.13 Create TAGs for all the Vehicles of a User

URL:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/createTagForUser/{user}/{expiry}?emails=bat@geotrackers.com>

Params:

user: user for which tags are created. The user should belong to the same organization as the web-service user

Rest of the parameters are same as other TAG APIs

Purpose:

Bulk create TAGs for all the vehicles of a user

Example:

<http://blazer7.geotrackers.co.in/GTWS/gtWs/LocationWs/createTagForUser/bat/20161231120000?emails=bat@geotrackers.com>

Response:

```
{
  "status": "success", "tag": "1666", "vehicle": "Ganesh_X1_V6", "expiry": 1480483440000,
  "status": "success", "tag": "8875", "vehicle": "rahul_temp2", "expiry": 1480483440000,
  "status": "success", "tag": "7208", "vehicle": "Devyat_X1", "expiry": 1480483440000,
  "status": "success", "tag": "9458", "vehicle": "PuneX1", "expiry": 1480483440000,
  "status": "success", "tag": "7898", "vehicle": "Devyat_X8", "expiry": 1480483440000,
  "status": "success", "tag": "2265", "vehicle": "Nandurbarcar1", "expiry": 1480483440000,
  "status": "success", "tag": "9383", "vehicle": "Delhi_GT", "expiry": 1480483440000,
  "status": "success", "tag": "4968", "vehicle": "MumbaiX1", "expiry": 1480483440000,
  "status": "success", "tag": "1680", "vehicle": "rahul_car_temp", "expiry": 1480483440000,
  "status": "success", "tag": "9865", "vehicle": "DelhiX1", "expiry": 1480483440000,
  "status": "success", "tag": "9391", "vehicle": "rahul_car1", "expiry": 1480483440000,
  "status": "success", "tag": "8423", "vehicle": "Delhi_GT_Dev", "expiry": 1480483440000,
  "status": "success", "tag": "8596", "vehicle": "testCar", "expiry": 1480483440000,
  "status": "success", "tag": "5364", "vehicle": "Ganesh_x1_01", "expiry": 1480483440000,
  "status": "success", "tag": "4397", "vehicle": "rahulTest Car", "expiry": 1480483440000,
  "status": "success", "tag": "2474", "vehicle": "Pune_PTR", "expiry": 1480483440000,
  "status": "success", "tag": "1902", "vehicle": "mumbaiCar", "expiry": 1480483440000,
  "status": "success", "tag": "1714", "vehicle": "Delhi_GT_104", "expiry": 1480483440000,
  "status": "success", "tag": "6769", "vehicle": "Delhi_PTR", "expiry": 1480483440000,
}
```

```
{"status":"success","tag":"3967","vehicle":"Nandurbarcar2","expiry":1480483440000},  
{"status":"success","tag":"4334","vehicle":"Local_PTR","expiry":1480483440000}
```