

Wialon TollTally Initial Integration	2
Brief Overview	2
Brief Background	2
Useful Links and Resources	2
Credentials & Developer Access	2
POCs and Contacts	3
Integration Scope	3
Frontend Tasks	3
Backend Tasks	3
Task Breakdown & Timeline	3
Risks and Dependencies	4
Deliverables	4

Wialon TollTally Initial Integration

Brief Overview

The primary goal is to get listed on the Wialon Marketplace, unlocking visibility to their partner ecosystem and over 4M+ connected vehicles.

This will be a 2-week effort:

- Week 1 to 1.5: Tech team to complete the integration (OAuth login, fetch vehicle/trip data, build sync endpoints).
- Remaining time: Marketing team to handle Marketplace listing, documentation, branding, and communication with Wialon.

Brief Background

Wialon (by Gurtam) is a global leader in GPS tracking with 4.1M+ connected devices. Partnering with Wialon through the Marketplace allows MapUp to:

- Instantly gain visibility with Wialon's 2,700+ telematics partners worldwide.
- Enable Wialon fleet customers to connect to TollTally with minimal friction.
- Validate our TollTally offering for global expansion and future co-selling.

Useful Links and Resources

Resource	Link
Wialon SDK/API Docs	https://help.wialon.com/en/wialon-hosting/user-guide/sdk
OAuth Token Login	https://hosting.wialon.com/login.html?access_type=-1&duration=0
Session (SID) Auth Guide	https://wialon-help.link/edda3211
Wialon Marketplace	https://wialon.com/en/marketplace
API Example Webinar	YouTube Video

Credentials & Developer Access

- Test Account URL: <https://hosting.wialon.com>
- Username: mapup_free
- Password: Tolltally@12345

- Auth Notes:
 - Use OAuth to generate access_token
 - Use token to fetch session ID (eid) from Wialon API

POCs and Contacts

Name	Role	Contact
Akash Gachinmani	Project Owner, External Comm	Akash Gachinmani
Anshul Kasana	Integration Lead	Anshul Kasana
Divyansh Sareen	Sub Lead and Backend POC	Divyansh Sareen
Atharv Diwan	Backend Developer	Atharv Diwan
Valery Lobatsevich	Technical Growth, Wialon	v.lobatsevich@wialon.com
Olga Galai	Partner Success, Wialon	o.galai@wialon.com
Iryna Syrnikava	Technical Care, Wialon	support@wialon.com

Integration Scope

Frontend Tasks

- ☐ Add Wialon to “Connect Integration” modal (OAuth-based).
- ☐ Show Wialon connection status and vehicle/trip previews.
- ☐ Handle loading/error states.

Backend Tasks

- ☐ Implement OAuth token → session ID flow.
- ☐ Securely store session credentials.
- ☐ Build vehicle and GPS data sync from Wialon API.
- ☐ Normalize schema to TollTally standards.

Task Breakdown & Timeline

Task	Details	Status	ETA	Notes
------	---------	--------	-----	-------

Obtain developer credentials	Request and acquire test credentials for Wialon platform access	Completed ▾	May 15	05/16/2025: - Login shared by Valery (mapup_free / !MapUp_1)
GPS tracking fetch	Fetch trip history or live tracking data for demo/test vehicles	Blocked ▾	TBD	
OAuth token login setup	Implement OAuth login flow to obtain access_token and session ID (eid)	Not Started ▾	May 17	
Vehicle list fetch	Use Wialon API to fetch list of vehicles for authenticated user	Not Started ▾	May 18	
Modal integration (UI)	Add Wialon to the "Connect Integration" modal with OAuth trigger	Not Started ▾	May 20	
Data preview UI	Display fetched vehicle/trip data to user post-authentication	Not Started ▾	May 21	
Sync normalization	Map Wialon data structure to TollTally vehicle and toll event schema	Not Started ▾	May 22	
Internal QA & validation	Run integration tests with test accounts and mock vehicle data	Not Started ▾	May 23	

Risks and Dependencies

- ☐ Awaiting GPS tracking test data for demo vehicles.
- ☐ Wialon's API is XML-heavy; requires careful parsing.

Deliverables

- ☐ Basic but functional Wialon Integration in TollTally (OAuth + vehicle + GPS ingestion).
- ☐ UI Modal integration with status, error handling.
- ☐ Verified with test vehicles.
- ☐ Marketing materials prepared for Marketplace.
- ☐ Wialon notified for Marketplace listing.

Technical Information

To get the GPS track and Timestamps we will be using this API

[API](#) ->

filter	filter, search in "pos" and "p"
--------	---------------------------------

We need to pos and t it has timestamp in epoch format we need to convert in our format

```
"pos": {
  "y": 53.84541,
  "x": 27.4470783333,
```

Here we can pass a flag which will remove the empty pos objects
This is the sample curl and sampe response

```
None
curl --location
'https://hst-api.wialon.com/wialon/ajax.html?svc=messages%2Fload_interval' \
--header 'accept: */*' \
--header 'content-type: application/x-www-form-urlencoded' \
--header 'origin: https://hst-api.wialon.com' \
--header 'referer: https://hst-api.wialon.com/wialon/post.html' \
--header 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36' \
--data-urlencode
'params={"itemId":29299852,"timeFrom":1747299311,"timeTo":1747817469,"flags":1,
"flagsMask":1,"loadCount":4294967295}' \
--data-urlencode 'sid=02b189cbd09005e08c3923a0ae8a658d'
```

For unit

For units first we have to make 2 api calls one to get all the units and one to get the details about the individual unit. Like vin and weight and co2 emission .

For getting all the units in the same api call just pass the flag as 1 .

Attached the sample curl with a sample response

None

```
curl --location --globoff --request POST
'https://hst-api.wialon.com/wialon/ajax.html?svc=core%2Fsearch_items&params={%2
2spec%22%3A{%22itemsType%22%3A%22avl_unit%22%2C%22propName%22%3A%22sys_name%22%
2C%22propValueMask%22%3A%22*%22%2C%22sortType%22%3A%22sys_name%22}%2C%22force%2
2%3A1%2C%22flags%22%3A1%2C%22from%22%3A0%2C%22to%22%3A0}&sid=023fcacb026ae58c40
bec671eb45909b'
```

response-:

We get this sample:

JavaScript

```
"items": [
  {
    "nm": "001_NO_TOLL",
    "cls": 2,
    "id": 29299852,
    "mu": 0,
    "uac1": 4178867978239
  }
]
```

In this items array we would be getting array of objects and we have a common field as id and on the basis of that we can make an individual call getting us to know everything about that . i think we would need this id and this nm (name of vehicle)