splyt

Senior Software Engineer Technical Task

You are part of the backend team responsible for building the core real-time services of a ride-hailing platform. A critical requirement is to **track the live location of drivers**. As part of this effort, you are tasked with designing and implementing a microservice responsible for:

1. **Receiving high-frequency location updates** from drivers' mobile devices
2. **Storing and maintaining** the most recent driver location
3. (Optional) Supporting **retrieval of historical movement** data

This microservice will serve as a foundational component of a ride-hailing platform and is expected to be **efficient, consistent, and production-ready**.

Task One — Process and Publish Driver Location Data

To begin, please process the file provided: `driver_location_log.json`. This file contains simulated driver location updates with the following fields:

Field	Description
`driver_id`	Id which uniquely identifies a driver
`latitude`	Latitude of the driver at that point of time
`longitude`	Longitude of the driver at that point of time
`time_offset_sec`	The number of seconds after the start of the script (**T_0**) when the driver location is published.

Your task is to **write a script** that:

1. **Reads** the `driver_location_log.json` file
2. **Sequentially and synchronously send** updates at the appropriate time (relative to when the script is executed, **T_0**) to the ingestion endpoint (implemented in Task Two)

Task Two - Implement the Driver Location Ingestion Service

Implement the location ingestion service which will receive frequent location updates from drivers (From Task One).

- Endpoint: **`POST: /location`** (*Does not need to be RESTful*)
- It should **ONLY** receive **`latitude`**, **`longitude`** and **`driver_id`** values sent from the drivers` phones.
- Be able to store the records effectively. (implemented in Task Three)

Task Three - Implement Storage and Query Capabilities

Your service must be able store the location updates for each driver and be able to return their **latest** known location. You should design a data model and storage strategy that:

- Efficiently track the latest location per driver
- Be able to handle high-frequency updates without performance degradation
- Prevents race conditions or data inconsistency due to out-of-order or duplicated requests
- Create a **GET** endpoint which takes in `driver_id` and returns the latest driver location
- Optional: Be able to provide historical tracking of the driver movement

Solution Expectations

1. Implement your solution to the task requirements above and showcase to us your abilities and experience!
2. Authentication features are optional but welcomed.
3. Submission:
    a. Zip of your solution repository or gitlab/github repository links.
    b. How long you have spent on this task.
    c. (Optional) Presentation for your solution.
4. You are allowed to use any libraries or frameworks (as long as it is justified)
5. Preferable technology stack to use is:
    a. Typescript
    b. MongoDB - For Task Three

FAQS

1. How long do I have to implement this tech task?

Our recommendation for this task is 2-3 days. In general, we do not set any time limits on our tech tasks. We encourage candidates to take their time to understand and to implement the tech task to the best of their abilities.

2. Can I expand on the tech task ?

Expansion and going beyond the requirements are always welcomed but are not necessary.

3. Can I use AI tools?

Yes, usage of AI tools to assist implementation is allowed. But candidates are expected to understand and should be able to explain the code submissions and concepts upon request..

4. Can I ask any questions ?

Yes, all questions are welcomed and clarifications can be requested for @ tech@splytech.io