DEEP
Consulting Solutions

Deep Consulting Solutions
600 Congress Ave, 14 Floor
Austin, TX, 78701
o: +1(512)387-9887
e: helpme@deepconsulting.solutions

Task 1

Problem

Build a metric logging and reporting service that sums metrics by time window for the most recent hour. You will build a lightweight web server that implements the two main APIs defined below.

APIs

POST metric

Request

```
POST /metric/{key}
{
  "value": 30
}
```

Response (200)

```
{}
```

GET metric sum

Returns the sum of all metrics reported for this key over the past hour

Request

```
GET /metric/{key}/sum
```

Response (200)

```
{
  "value": 400
}
```

Clarifications

- For the sake of the problem, persistence is not required. Therefore don't use a database but just use in-memory data structures or file storage only.
- You can use either Node standard library or small web framework like Express.
- You should optimize for both readability of your code and performance.
- All values will be rounded to the nearest integer.
- You need to get rid of any reported data after it is more than an hour old since we only need up to the most recent hour.
- The solution should be submitted as per instructions by the issuer in the form of a public GitHub repository.
- Applicable testing, logging and good documentation is expected.

Example

Imagine these are the events logged to your service for a metric "active_visitors"

```
// 2 hours ago **
POST /metric/active_visitors { "value" = 4 }

// 30 minutes ago
POST /metric/active_visitors { "value" = 3 }

// 40 seconds ago
POST /metric/active_visitors { "value" = 7 }

// 5 seconds ago
POST /metric/active_visitors { "value" = 2 }
```

These are the results expected from calling get aggregates:

```
GET /metric/active_visitors/sum // returns 12
```

** Note that the metric posted 2 hours ago is not included in the sum since we only care about data in the most recent hour for these APIs.