# API:

**createMetric**

**HTTP POST**

**"/metrics"**

Runtime:

|  |
| --- |
| Best case scenario: when user not send any MetricUnit: O(1) |
| Worst case scenario: when user send n metricunits: O(nlogn)  Since the activity of adding 1 MetricUnits cost O(logn) because of the calculating Median function, each adding cost O(logn) to get the new median  Therefore with n element, it takes O(nlogn) |

# API:

**getMetric**

**HTTP GET**

**"/metrics/{id}"**

Runtime:

|  |
| --- |
| O(1) |
| Since the in memory storage using hash map, which take O(1) to get an element |

# API:

**getStatistic**

**HTTP GET**

**"/metrics/{id}/statistics"**

Runtime:

|  |
| --- |
| O(1) |
| Since all the statistics have been calculated on the insert and create operation, to get a specific statistic object, it cost O(1) for the in memory hash map to call get method |

# API:

**addMoreValues**

**HTTP POST**

**"/metrics/{id}/metricUnits"**

Runtime:

|  |
| --- |
| O(nlogn) |
| Since the activity of adding 1 MetricUnits cost O(logn) because of the calculating Median function, each adding cost O(logn) to get the new median  Therefore with n element, it takes O(nlogn) |