

Coding Challenge

We would like you to implement a simple AdTech web service that tracks the ads that are being delivered through their lifecycle and generates some simple statistics. You are free to use any tech stack you prefer. The code should be easy to run, preferably in a docker container.

We are mostly interested in how you create and structure the code, not necessarily in any performance aspects. As we do not expect a production-ready solution, we would also like to know how you would design a highly-scalable, highly-available system that performs similar tasks (assume more fields for each message, more complicated statistics to be queried etc).

Please write a couple of paragraphs explaining which data systems you would use and why and how you would handle scaling on the service side. A block diagram showing the data flow through the pipelines is much appreciated

You should allocate a couple of hours whenever you have some time off to complete the project.

Helpful take-home project guidelines:

- This project will be used to evaluate your coding and system design skills. We are interested in the programming patterns that you use and the quality of the code and less interested in any performance aspects.
- Please keep it simple and feel free to use any database engine(in-memory, persistent, no-database at all) that takes the least amount of time to implement for you. However, do take time to write the high-level system design document.

Ingestion

POST /ads/delivery that is triggered whenever one of our ads is loaded on a page and follows the spec below:

Request format:

```
{  
  "advertisementId": 4483,  
  "deliveryId": "244cf0db-ba28-4c5f-8c9c-2bf11ee42988",
```

```
"time": "2018-01-07T18:32:23.602300+0000",  
"browser": "Chrome",  
"os": "iOS",  
"site": "http://super-doooper-news.com"  
}
```

Response format:

- HTTP response code 200 if all went fine
- HTTP response code 500 if any error occurred

POST /ads/click that is triggered whenever an user clicks on one of the ads we delivered.

It follows the spec below:

```
{  
  "deliveryId": "244cf0db-ba28-4c5f-8c9c-2bf11ee42988",  
  "clickId" : "fff54b83-49ff-476f-8bfb-2ec22b252c32",  
  "time": "2018-01-07T18:32:34.201100+0000",  
}
```

Response format:

- HTTP response code 200 if everything went fine
- HTTP response code 404 if we never received the given delivery
- HTTP response 500 if something went wrong

POST /ads/install that follows the spec below:

```
{  
  "installId": "144cf0db-ba28-4c5f-8c9c-2bf11ee42988",  
  "clickId" : "fff54b83-49ff-476f-8bfb-2ec22b252c32",  
  "time": "2018-01-07T18:32:34.201100+0000",  
}
```

Response format:

- HTTP response code 200 if everything went fine
- HTTP response code 404 if we never received the given click
- HTTP response 500 if something went wrong

Statistics

```
GET /ads/statistics/time/{start}/{end}/overall
```

Description: Returns the number of views, clicks and installs in the given period

Example: `/ads/statistics/time/2018-01-07T14:30:00+0000/2018-01-07T18:20:00+0000/overall`

Response format:

```
{
  "interval" : {
    "start" : "2018-01-07T14:30:00+0000",
    "end" : "2018-01-07T18:20:00+0000"
  },
  "stats": {
    "deliveries": 10,
    "clicks": 4,
    "installs": 1
  }
}
```

```
GET /ads/statistics/time/{start}/{end}/{category1}/.../{categoryN}
```

Description returns the number of views, clicks and installs in the given period grouped by the given categories

Example: `/ads/statistics/time/2018-01-07T14:30:00+0000/2018-01-07T18:20:00+0000/browser`

Response format:

```
{
  "interval": {
    "start": "2018-01-07T14:30:00+0000",
    "end": "2018-01-07T18:20:00+0000"
  }
}
```

```

},
"data": [
  {
    "fields": {
      "browser": "Chrome"
    },
    "stats": {
      "deliveries": 10,
      "clicks": 4,
      "installs": 1
    }
  },
  {
    "fields": {
      "browser": "Safari"
    },
    "stats": {
      "deliveries": 2,
      "clicks": 1,
      "installs": 1
    }
  }
]
}

```

Example: `/ads/statistics/time/2018-01-07T14:30:00+0000/2018-01-07T18:20:00+0000/
browser/os`

Response format:

```

{
  "interval": {
    "start": "2018-01-07T14:30:00+0000",
    "end": "2018-01-07T18:20:00+0000"
  },
  "data": [
    {

```

```
    "fields": {
      "browser": "Chrome",
      "os": "Android"
    },
    "stats": {
      "deliveries": 10,
      "clicks": 4,
      "installs": 1
    }
  },
  {
    "fields": {
      "browser": "Safari",
      "os": "iOS"
    },
    "stats": {
      "deliveries": 2,
      "clicks": 1,
      "installs": 1
    }
  },
  {
    "fields": {
      "browser": "Chrome",
      "os": "iOS"
    },
    "stats": {
      "deliveries": 15,
      "clicks": 5,
      "installs": 3
    }
  }
]
}
```