

Backend Engineer Code Test

GraphQL API Server

Requirements:

- Language: Node.js
- Library: Apollo Server ([A Introduction to Apollo Server](#))

Create an GraphQL API Server that supports the provided schema and use the provided JSONs as the datasource. The API should be authenticated using **Bearer** token.

Please provide the token in your submission.

Schema

```
1  type Action {
2    _id: ID!
3    createdAt: Long!
4    updatedAt: Long
5    name: String!
6    description: String
7    functionString: String
8    resourceTemplateId: ID
9    resourceTemplate: ResourceTemplate
10 }
11
12 type Trigger {
13   _id: ID!
14   createdAt: Long!
15   updatedAt: Long
16   name: String!
17   description: String
18   functionString: String
19   resourceTemplateId: ID
20   resourceTemplate: ResourceTemplate
21 }
22
23 type Response {
24   _id: ID!
25   createdAt: Long!
26   updatedAt: Long
27   name: String!
28   description: String
29   platforms: [ResponsePlatform]
30 }
31
32 type ResponsePlatform {
33   integrationId: ID
34   build: Int
35   localeGroups: [ResponseLocaleGroup]
36 }
37
38 type ResponseLocaleGroup {
39   localeGroupId: ID
40   variations: [ResponseVariation]
```

```
41 }
42
43 type ResponseVariation {
44     name: String!
45     responses: JSON
46 }
47
48 type ResourceTemplate {
49     _id: ID!
50     createdAt: Long!
51     updatedAt: Long
52     name: String!
53     description: String
54     schema: JSON
55     integrationId: String
56     functionString: String
57     key: String
58 }
59
60 type NodeObject {
61     _id: ID!
62     createdAt: Long!
63     updatedAt: Long
64     name: String!
65     description: String
66     parents: [NodeObject]
67     parentIds: [ID]
68     root: Boolean
69     trigger: Trigger
70     triggerId: ID
71     responses: [Response]
72     responseIds: [ID]
73     actions: [Action]
74     actionIds: [ID]
75     priority: Float
76     compositeId: ID
77     global: Boolean
78     colour: String
79 }
80
81 type Query {
82     node(nodeId: ID): NodeObject
83 }
```