

Take-home project

Oct 2018 - Dinopark Ops

Note: This take-home challenge aims to assess the technical ability of candidates applying for a full-stack software engineering position. Completing this exercise should demonstrate that you can:

- 1. Understand and interpret requirements*
- 2. Demonstrate proficiency in a programming language*
- 3. Build a reactive, web-based user-interface*
- 4. Write clean, well-structured code*

The deliverables are a ZIP file containing:

- 1. The created project/code*
- 2. readme.md file answering the following:*
 - a. How to set up and run the code*
 - b. Briefly explain how you approached the problem*
 - c. What you would do differently if you had to do it again*
 - d. What you learned during the project*
 - e. How you think we can improve this challenge*

This challenge should ideally take 2-4 hrs, but as it is a new exercise, actual completion times may vary.

PS: please do not share this challenge with anyone.

Brief

Dinopark covers an expansive area. As such maintenance of facilities and security features (fences, cameras, etc.) are a constant effort. Here at park maintenance, we take our jobs very seriously. If we didn't, someone would get hurt! We need you to create a tool to help make our jobs safer and more efficient. Last year we lost eight workers. We really want to get that number down.

Let me tell you a little bit more about our operations:

- Our team divides the park into a 26 by 16 grid of zones. The letters A-Z represent the columns and the numbers 0-15 represent the rows. For example, A13 is the first column and the 14th row.
- Each zone in the grid needs maintenance every 30 days.
- Maintenance staff can only perform maintenance in a zone when it is *safe* to do so.
 - *Safe* means: No carnivores are in a zone or all carnivores in the zone are still digesting their last meal (all dinosaurs in the park have an estimated digestion time which is logged when the animal is first moved into the park)

We'd like you to build a user-interface which processes logs from our park's monitoring system (NUDLS™), determining and indicating which zones, at time of viewing:

- Need maintenance (gear icon). Ignore zones that have never been maintained before.
- And are safe to enter (#51B687 for *safe*, red *unsafe*, and white otherwise)

NUDLS™ offers an endpoint which returns a list of logged activity. I've also attached the NUDLS™ developer docs (such that they are).

<https://dinoparks.net/nudls/feed>

We've mocked up a desired design for the user-interface. The resources are available as PNG's here: [user-interface](#), [logo](#) and [wrench](#).

NUDLS (New Unified Dinopark Logging System) Developer Documentation

Version 1.4.7-a5, build 10

NUDLS is a system for allowing employees of Dinoparks Amusement and Betting Company GMBH access to an unfiltered view of park operations.

NUDLS uses the new JSON (JavaScript Object Notation) spec which was just ratified.

NUDLS uses the idea of a global persistent log as a unifying abstraction. After adding a new system, NUDLS sends the newly integrated system all previously persisted events one by one. NUDLS uses the HTTP (Hypertext Transfer Protocol) as a transport mechanism, sending logged messages using a POST (Personal Object Store Transfer) request.

NUDLS uses ISO standards for all datetime needs.

The body of this request may be any one of the following kinds (along with a sample message attached)

Dino Added

This event is called when new animals are added to the park. Digestion period should be used to inform feeding schedule.

```
{
  "kind": "dino_added",
  "name": "McGroggity",
  "species": "Tyrannosaurus rex",
  "gender": "male",
  "id": 1032,
  "digestion_period_in_hours": 48,
  "herbivore": false,
  "time": "2018-04-20T11:50:53.460Z",
```

```
"park_id": 1
```

```
}
```

Dino Removed

This event is called when an animal is removed from the park

```
{
```

```
"kind": "dino_removed",
```

```
"id": 1032 ,
```

```
"time": "2018-04-20T11:50:53.460Z",
```

```
"park_id": 1
```

```
}
```

Dino Location Updated

This event is called when an animal moves to a different cell in a standardized park grid.

```
{
```

```
"kind": "dino_location_updated",
```

```
"location": "E10",
```

```
"dinosaur_id": 1032,
```

```
"time": "2018-04-20T11:50:53.460Z",
```

```
"park_id": 1
```

```
}
```

Dino Fed

Indicates when a dinosaur was fed

```
{
```

```
"kind": "dino_fed",
```

```
"id": 1032,
```

```
"time": "2018-04-20T11:50:53.460Z",
```

```
"park_id": 1
```

```
}
```

Maintainence Performed

Indicates that a routine maintenance pass was made at the given cell in the standardized park grid. Please see Winged Avengers: The Dinopark Employee Handbook and Survival Guide, Page 107 for full details on what routine maintenance activities entail and how it may affect **you**

```
{  
  "kind": "maintenance_performed",  
  "location": "E7"  
  "time": "2018-04-20T11:50:53.460Z",  
  "park_id": 1  
}
```