BOB.IO CODING CHALLENGE

Welcome to Bob coding challenge!

This coding challenge has been made in order to test your Full Stack developer skills. Our stack is **React+Redux** in the frontend and **NodeJS** and **MongoDB** in the backend. If you want to add a CSS preprocessor, we are using **JSS**. For testing, we use **Mocha** and **Chai** in the backend.

From the time we send you this PDF, you will have 4 hours to try to finish all the requirements. Don't worry if you don't have enough time to finish it, but the more you do, the better knowledge we will have!!

Once those 4 hours are passed away (or even before if you want), let us know and send us your solution. We will have a small call if you want to explain us something about the code you just do, what do you think about this test and how was the process.

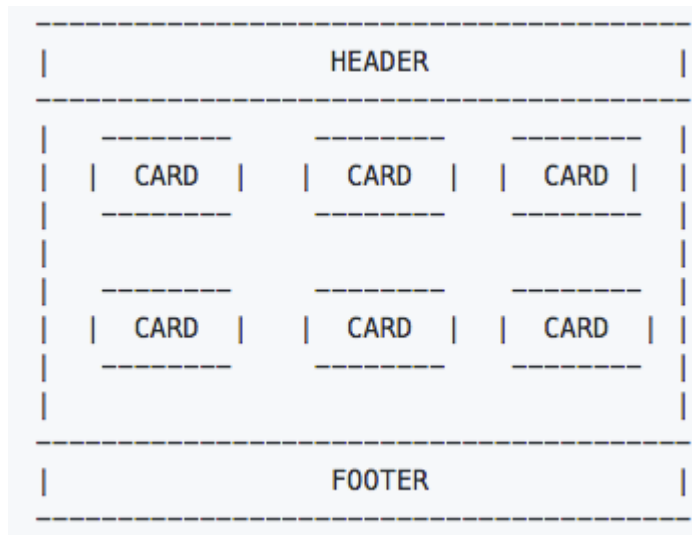Our test is divided in three exercises.

Backend

Create a **NodeJS**, **Express** and **MongoDB** project in order to create an API that can create, read, update and delete this kind of collection:

```
{
  Name:   'John Doe',
  Bags: 5
}
```

This is the minimum expected, you can add more data if you want!! - **Testing is really appreciated**, we try to do it in all our code :-)

Frontend Components Creation and Printing

Fetch the data from the API you just created, and print it to a cool home site.

```
------------------------------------------------
|                  HEADER                       |
------------------------------------------------
|     ----------      ----------      ----------    |
|    |  CARD  |    |  CARD  |    |  CARD  |   |
|     ----------      ----------      ----------    |
|                                                |
|     ----------      ----------      ----------    |
|    |  CARD  |    |  CARD  |    |  CARD  |  |
|     ----------      ----------      ----------    |
|                                                |
------------------------------------------------
|                  FOOTER                        |
------------------------------------------------
```

Each card should show the name and the bags of each passenger. If you want to add more data, feel free!

Create a link in the header for the next section.

- Extra: Create a pagination after the 3rd step.

Frontend SPA with data entry

Create a new route for the app you created. That route should be named /new and make an API call to create a new collection.
In order to do that, you will have to create three components:

1. A bag selector. Allowing not more than 5 bags.

2. A text input. Would be great if can handle the state of "active", "dirty" and "error". The name must start with uppercase and have two or more words. If not, it will show an error message.
3. A button for submitting. Enabled or disabled depending of the possibility of creating or not a new customer.

Pieces of advice

We will like your app if:
- You follow clean coding practices
- You test all the app
- You use git properly
- You think about the UX a little bit

Let us know if you have any question or feedback.
Hope to see your code soon!