silicon rhino

Engineer Technical Assessment

1.OVERVIEW

We would like you to create a web application which retrieves data from the API of the *drinks* app, and showcases the events, however you find most convenient. This should be in Javascript or Typescript (with any framework you wish), and you should go to as much detail as you have time for.

This is not exclusively a test of front end design skills but we would want the application to look as attractive as possible given the time provided.

2. API & DATA STRUCTURE

The REST API provides programmatic access to read *drinks* data. The REST API doesn't make use of any authentication. Responses are in JSON.

RESOURCES

/events [GET] - returns a collection of events matching the parameters values

Parameters:

```
page: page number to return
pageSize: size of page (number of events per page)
search: search query term (this is matched against the title of events and name
of the location)
```

Example:

```
https://mock-api.drinks.test.siliconrhino.io/events?page={pageNumber}&pageSize=
{pageSize}&search={searchTerm}
```

/event/{id} [GET] - returns a single event

Parameters:

id: id of the event to return

Example:

```
https://mock-api.drinks.test.siliconrhino.io/events/{id}
```

AVAILABLE INTERFACES

The typescript interfaces are as follows:

```
interface Event {
  id: number;
  time: string;
  title: string;
  creator: User;
  guests: User[];
  type: 'BEERS' | 'COCKTAILS' | 'COFFEES' | 'MILKSHAKES';
  location: EventLocation;
  comments: EventComment[];
}

interface EventLocation {
  name: string;
  latitude: number;
  longitude: number;
}

interface EventComment {
  user: User;
  timestamp: string;
  message: string;
}

interface User {
  name: string;
  avatarUrl: string;
}
```

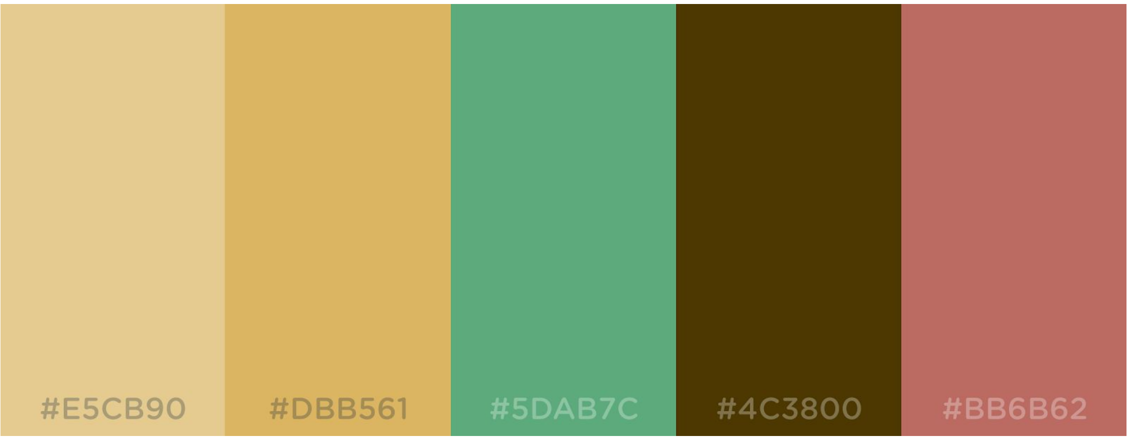
3. TECHNOLOGY

The system can use any server-side technology of your choosing (if required), and should use a javascript based front-end (e.g. AngularJS). We are happy for you to utilise any base templates or tools for Rapid Application Development as well as any CSS frameworks for styling.

4. MATERIALS

You have access to a .ZIP file with all the relevant icons and logos to use at your discretion.

Drinks color palette: https://coolors.co/e5cb90-dbb561-5dab7c-4c3800-bb6b62



5. REQUIREMENTS

This task should take you between 2 and 4 hours, and you should make sure the code showcases your skills in the best way possible. Please provide a link to a code repository with your implementation.

6. TIPS

- Use a CSS preprocessor
- There will likely be multiple views/pages within the overall application
- Think about interesting ways to make use of the co-ordinates data
- Don’t forget about mobile browsers
- You might implement a caching layer (e.g. API + database)