



GOHIGHLEVEL BACKEND DEV HIRING CHALLENGE

Problem statement :-

The purpose of this tool (GHL) is to make appointment booking between people a lot easier. The usual process is to exchange mails between people till they find a convenient time.

Here is the test task :-

You need to create the backend project with node.js/express.js with firestore integration for data storage and create the private git repository on github.

Purpose of this task is to assume you are building a calendar appointment system where users can see free slots of Dr. John and they can book for whatever time period when Dr. John is available, which is already a big module of our app, For reference and example can visit this link

You will need to create one collection in the firestore `events` which will hold all the events and based on events we can decide at what time you are free.

Create these static config variable in app :- (You can decide the variable type)

1. Start Hours - which will suggest at which time you want to start your availability.
2. End Hours - which will suggest at which time you want to end your availability.
3. Duration - Duration of slot
4. Timezone - America/Los_Angeles

Let's say I set my availability from 10AM - 5PM and slot Duration 30 minutes so free slot API you are supposed to return all the available slots, which will suggest that any customer can book Dr. John at any of these times.

Example Output (incase we don't have any event)

```
[  
  '2019-11-14T10:00:00',  
  '2019-11-14T10:30:00',  
  '2019-11-14T11:00:00',  
  '2019-11-14T11:30:00',  
  ....  
  '2019-11-14T16:30:00', // as the your availability is upto 5PM]
```

In case I have an event already at 2019-11-14T10:00:00, that slot should be excluded.

These are list of the APIs

1. Free Slots:-

Params:

- Date
- Timzone

Return all the free slots available for a given date converted to whatever timezone we pass.

Example:-

By default the free slots will be like 10AM, 10:30AM, 11:00 AM.. so on, but that is as per the US/Eastern (or whatever you will set in config/const)

So it simply suggests Dr John. is available from 10AM-5PM as per US/Eastern timezone. Now if someone from India wants to book Dr. John they will like to see Dr. John timing as per IST (GMT+5:30)

So in this API you will pass Date and timezone in which you want to see Dr. John's availability, which will be like these in IST (GMT+5:30). You can use this inorder to help you test the timings : <https://www.thetimezoneconverter.com>

10:30 PM

11:00 PM

...

2. Create event:-

Params:

- DateTime (You can decide the format, timestamp or date format upto you)
- Duration (In minutes, INT)

Whatever data you will pass it will create the event and store that into the firestore document, if the event already exists for that time you need to return status code 422 or else just store it and return with status 200.

3. Get Events:-

Params:

- StartDate
- EndDate

Return all the events between given StartDate & EndDate

Optional UI:-

If you finished the above task and are able to create the example UI for the same in vue.js it will be a bonus to showcase full stack dev skills.

Create 2 page application

1. Book Event

- Add datepicker (to choose date)
- Input (to add minute duration)
- Dropdown to pick timezone (can keep 4-5 limited option)
- Get Free Slots Button

On that button click, fetch the free slots and show them as buttons or labels on UI and on click of one of them create an event with that time.

2. Show events

- Add date range picker

On selecting any date range show all the events.

Reference:-

1. This is an actual example link of our booking widget, if you want to know what you are building as tests.
<https://link.gohighlevel.com/widget/booking/9n7WVCGQSzkvOMRiMIhL>
2. Example test cases (contains the example test case without timezone calculation to understand how it should work)
https://docs.google.com/spreadsheets/d/1t7g3VRN-glGiNbZ4dMz4FS_IU2uS_rGtb6otxUvjXVs/edit#gid=0

Feel free to ask any further details or if require any further details on this demo task.

All the very best!