EY Selection test – Full stack developer

* ***Muruganantham (Anand) Sadasivam***

## Local setup of the project:

* Local setup of the project.
* Gone through the readme file and tried to run the commands on the makefile
* The app server started on port 3000 and API on 4000

Problems

### Database:

1. Fixed the database with non-null and constraint checks for all columns
   1. Not null condition for all columns
   2. Username validation
   3. Location string validation
   4. Created and booked columns are set as the timestamp
   5. The id column is set as uuid;
2. Seed- file Generate more synthetic data to have more records in the DB.

### UI:

1. Able to create bookings in the past shows the error but updates the database with the booking and displays it on the list page.
2. Attempting to create multiple bookings for the same venue on the same date, not displaying a meaningful error message on the web page.
3. The Fetch of the get bookings API is incorrect, looping the call on the page infinitely. The dependency array is not used on the useEffect hook.
4. “Cannot book in the past!” and booking is created. Both messages are displaying for the current date. Not sure about the business requirement for current date booking, displayed meaningful error messages on the web page
5. Delete button is just triggering the API and not reloading the page and deleting it on the page.

Displayed meaningful message on the page after deletion.

1. Added a refresh button on the listing page, which reloads the page with the latest booking information.
2. Added a sorting, filter, and pagination button on the booking list page.
3. Created a new component bookingRow, which will display each row for the data received from the API.

### API:

1. Creating the bookings for any city or string (Eg: Chennai, India) via direct API hit(postman).
   1. Created a new API endpoint called getBookingLocations which returns the locations from an enum file.
   2. Validating the location field on the request body against the enums before updating the DB.
2. Implemented pagination on the getBookings list to display the page-wise records on the UI.
3. Preliminary Unit tests for all the endpoints
4. Fixed the error handling logics in Post and Get end points

### Known errors/things to improve

1. Present + 1-day error on the create page.
2. Pagination issue on the get Booking location.
3. Authentication mechanisms for UI and API, the application is not secure
4. Create a form for login on the front end and jwt or Oauth-based authorization/authentication for the API.
5. Clear Business requirement documentation for new features or existing feature improvement.
   1. Current day booking policy
   2. Errors or redirection of web pages for error handling.
6. Continuous integration and continuous deployment pipeline for the application
7. Different kinds of testing functional/integration, performance testing like Postman for api and cucumber for frontend, jMeter.