Front-End Implementation

Packages

This project utilized React+Type Script to implement all front-end pages. The basic components (table, list, etc.) are customized from MUI Libraries. And the charts(Bar, Pie, Line Charts) are drawn with Recharts API.

Connection with Server

All of the user inputs and request are gathered at the front end pages, and sent to a localhost port running the server by Axios. The logic of such process are described as follows:

1. Gather user input
   User input includes form values, button clicking events, date range selection, etc. These information are all gathered by the components of front-end.
2. Sending request to server
   After getting the user input, the corresponding component sends http request to the server:

```
await addLocation(postData, token, email)
        .then((res) => {
          setSuccessSnackbar(true);
          window.location.href = "/location";
        })
        .catch((error) => {
          setFailSnackbar(true);
          setErrorInfo(error.data);
        });
```

The request functions are all implemented in the folder 'src/services', providing a good practice of seperating functional procedures and GUI component:

```
export const addLocation = async (data: AddLocationData, token: string, email:string) => {
  const params = {
    location_street_num: data.streetNum,
    location_street_name: data.streetName,
    location_unit_number: data.unitNumber,
    location_city: data.city,
    location_state: data.state,
    location_zip_code: data.zipCode,
    square_feet: data.squareFeet,
    num_bedrooms: data.numBed,
    num_occupants: data.numOccupants,
    email:email
  }
  const config = {
    headers: {
      'Authorization': `Bearer ${token}`
    }
  };
  return axiosInstance.post(`${constants.ENDPOINT_LOCATION_URL}/add`, params, config);
}
```

3. Handling request and response with Axios