

Assignment

Task 1: Signup and Login API (Only Backend)

Description

Your task is to implement Signup and login with JWT (Access and Refresh tokens)

- User can **signup** new account with personal login & password
- User can **login** with personal login & password, server returns response with Access token and Refresh token.
- **Refresh** token helps to get new pair Access token – User now should use valid Access token to access resources – When the Access token is expired, user can't use it anymore. Validity of Access Token is 120 seconds.
- **Delete user** API for delete user from system.
- Refresh token helps to get new pair Access token.
- Stored all relevant details in MongoDB.
- Basic System design principles should be considered in implementation such as same Email is not allowed twice for signup.
- Task should be implemented in Typescript.

Task 2: Fetch API and Create Webpages (Only Frontend)

Description

This assignment is designed to test frontend programming skills and whether you can grasp a new programming language. It needs to be implemented using Angular.

Milestone 1

Fetch the response from below URL and show the data in list format (refer table below).

URL - https://uat.utopiatech.in:4520/panel/gettestlist?org_id=3

Request Type - get

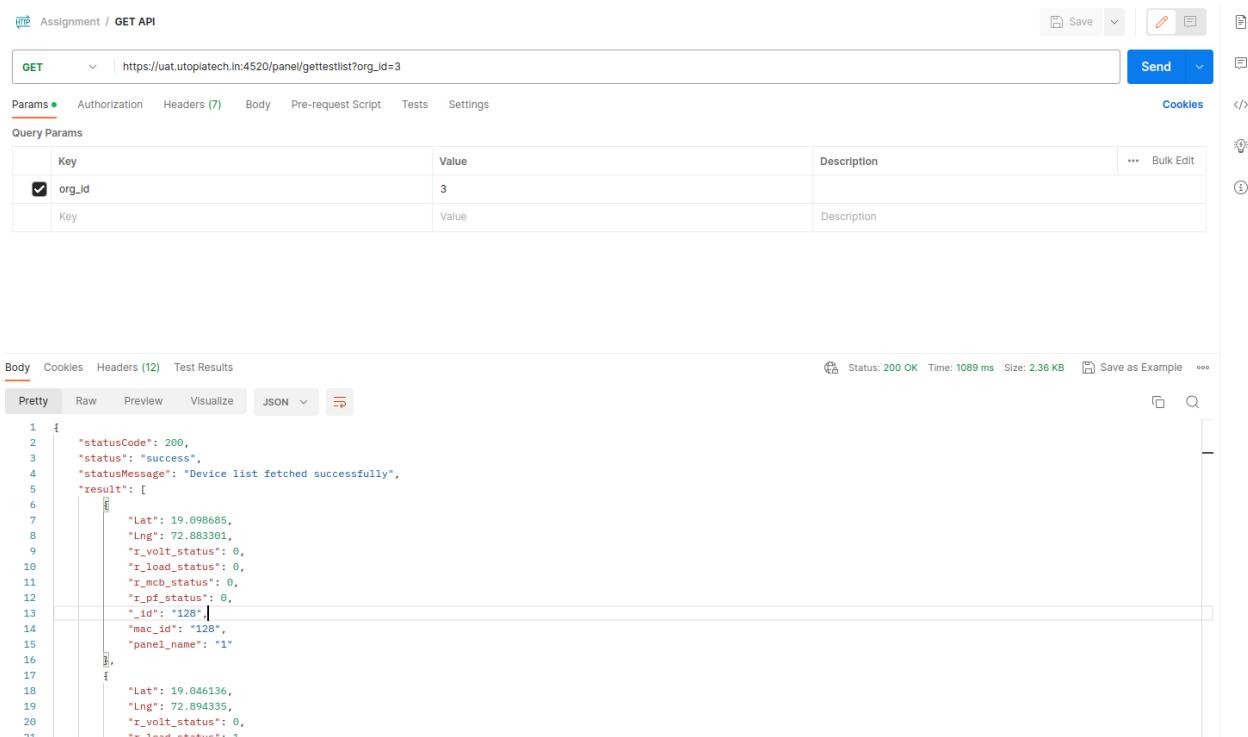
Params - org_id = 3Response -

```
{  
  "statusCode":200,  
  "status": "success",  
  "statusMessage": "Device list fetched successfully",  
  "result": [  
    {  
      "Lat": 19.098685,  
      ...  
    }  
  ]  
}
```

Table Format:

Panel Name	Mac Id	Lat	Long	Location
17000006	ABCD	19.098685	72.883301	Location_Icon
17000007	PORS	19.767478	72.956438	Location_Icon

API Call from Postman



The screenshot shows a Postman request for a GET API. The URL is `https://uat.utopiatech.in:4520/panel/gettestlist?org_id=3`. The 'Params' tab is selected, showing a query parameter `org_id` with value `3`. The response body is displayed in JSON format, showing a successful response with status code 200, status success, message Device list fetched successfully, and a result array containing two objects. Each object has properties like _id, mac_id, panel_name, Lat, Lng, r_volt_status, r_load_status, r_mcb_status, r_pf_status, and r_load_status.

```

1 {
2     "statusCode": 200,
3     "status": "success",
4     "statusMessage": "Device list fetched successfully",
5     "result": [
6         {
7             "_id": "128",
8             "mac_id": "128",
9             "panel_name": "1",
10            "Lat": 19.098685,
11            "Lng": 72.883301,
12            "r_volt_status": 0,
13            "r_load_status": 0,
14            "r_mcb_status": 0,
15            "r_pf_status": 0,
16            "r_load_status": 1
17        },
18        {
19             "_id": "129",
20             "mac_id": "129",
21             "panel_name": "2",
22             "Lat": 19.046136,
23             "Lng": 72.894335,
24             "r_volt_status": 0,
25             "r_load_status": 1
26         }
27     ]
28 }
  
```

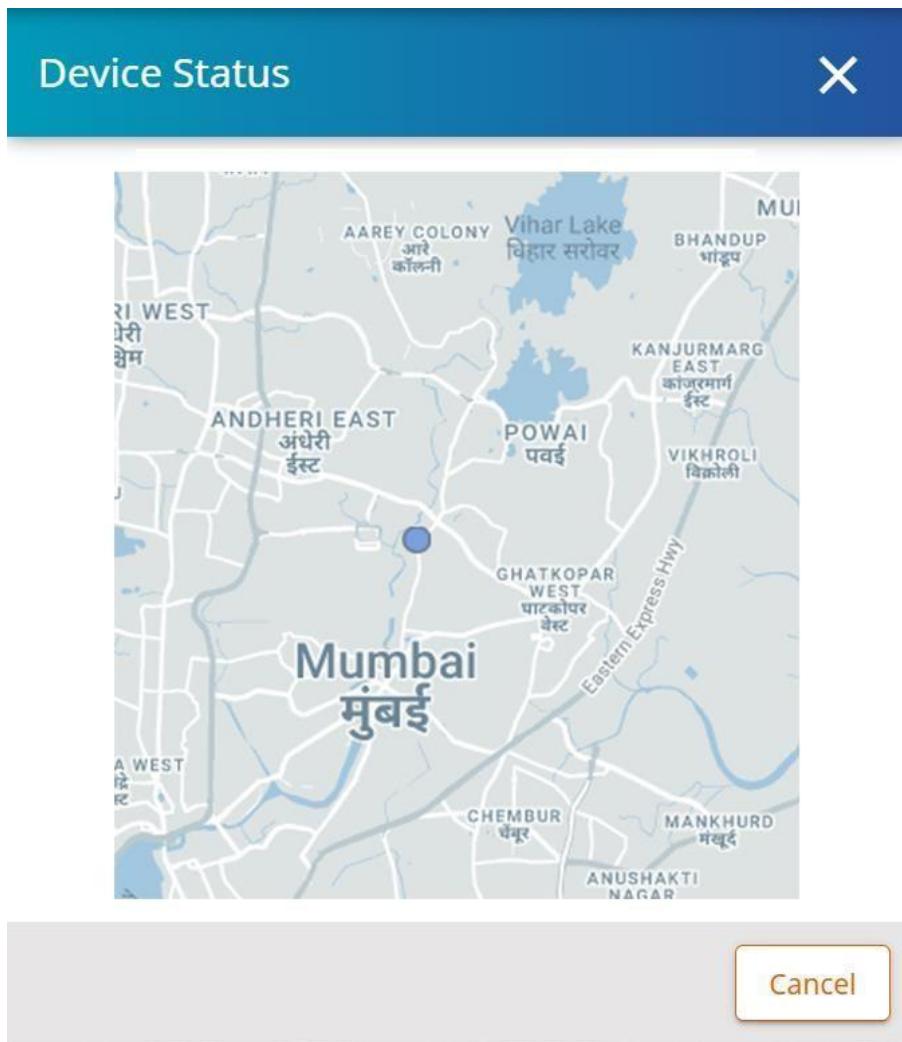
Milestone 2

If clicked on “Panel Name” (Table in Milestone 1) a Sidebar should open up. It should show details of panel in following format: -

Parameters	R Phase
Voltage Status	0
MCB Status	0
Load Status	1
PF Status	0

Milestone 3

- If clicked on Location Icon (Table in Milestone 1) a sidebar should open with map. Below is a reference Image –



- The location of the panel(marker) should be shown in the map
- On marker single click, latitude and longitude should be shown.