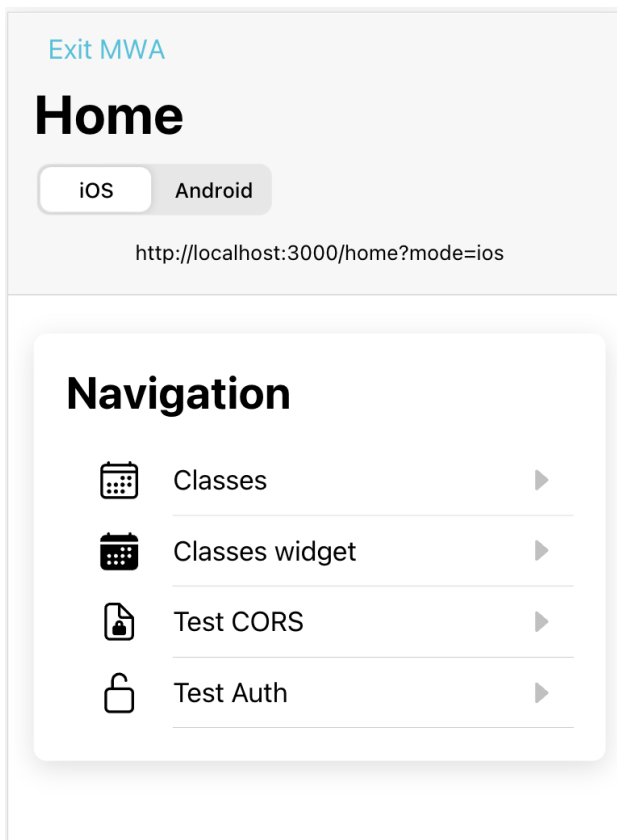


MWA Test Task

Implement two pages: a locations list page and a location details page. The locations list page should feature a functional search bar allowing users to quickly filter through the list of locations. If the search results in no matches, an appropriate empty state should be presented to the user. The location details page should include a toggle switch enabling users to alternate between displaying the location's working hours and contact information, including address, email, phone number, and website URL.

Extend [mwa-reference](#) app with new routes and add a link inside “Navigation” card which will lead to the locations list page.



Use the following public API to fetch the list of locations:

JavaScript

GET <https://egymidp.netpulse.com/np/company/children>

It returns the list of the following data type (some of the fields are omitted because they are not needed for the task). Put this type in the src/types and use while implementing the task.

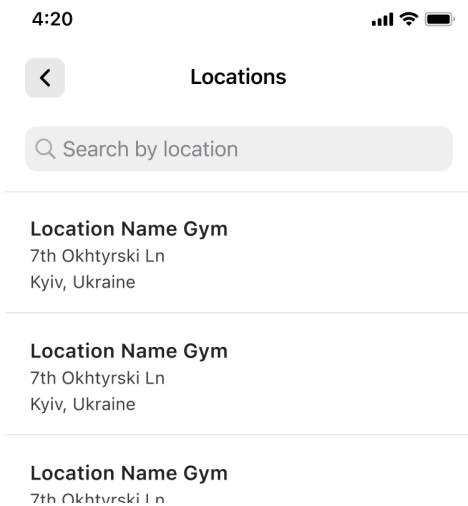
JavaScript

```
type Location = {
  uuid: string;
  name: string;
  timezone: string;
  phone: string;
  email: string;
  url: string;
  address: {
    addressLine1: string;
    addressLine2: string;
    city: string;
    country: string;
    postalCode: string;
    stateOrProvince: string;
    timezone: string;
    lat: number;
    lng: number
  },
  workingHours: {
    Mon: string;
    Tue: string;
    Wed: string;
    Thu: string;
    Fri: string;
    Sat: string;
    Sun: string;
  } | null,
  workingHoursFreeText: string | null,
},
```

It's preferred to follow the existing [mwa-reference](#) app structure on how to organize api calls, using react-query and hooks.

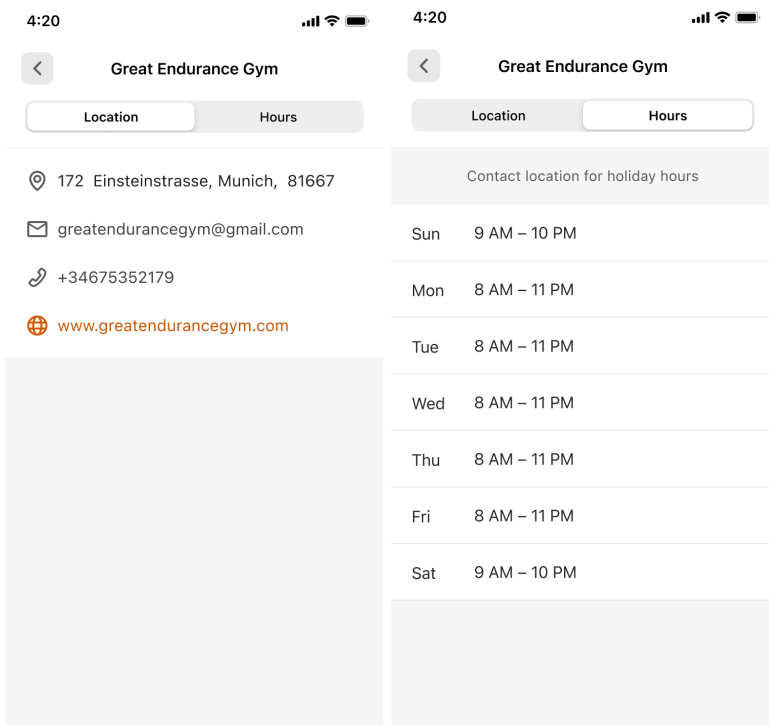
Locations list page:

- Clicking on back button opens the Home page
- Search by location name
- Empty state when no results found on search
- Click on list item leads to the location details page



Location details page:

- Clicking back button opens locations list page
- Add tabs to switch between location details and working hours view



Others:

- It's preferred to make texts lokalised
- We are not providing design files, like figma, but it would be preferred using the design system for typography and colors (add your css variables in variables.scss and custom classes), choose your font-sizes and colors close to the designs provided on images.
- Use simulator built from the ionic-cli while implementing
- Pages components should use corresponding native styles when opened on iOS and Android.