FED exercise

SmartGreen team Kiev

Goal

1

Build a SPA with few views, that are described in this document, based on json data, and ready for mobile and PC devices.

2

Keep code quality: Angular best practices, including unit-tests, and tool to run them.

3

Follow functional requirements described in this document, and feel free to design is according to your UI-UX recommendations.

4

Deliver back release that includes: minimized and obfuscated artifact hosted on web (+link to app), code maintained in GitHub project (+link to repo)

General requirements

Views:

- Sites list: list of buildings that are monitored by this app.
- Site overview: details of specific building's equipment
- Equipment analytics: details regarding specific equipment.
- Data is taken in real time from this JSON: http://www.smartgreen.co.il/fed/sitesdata.json

UI-UX:

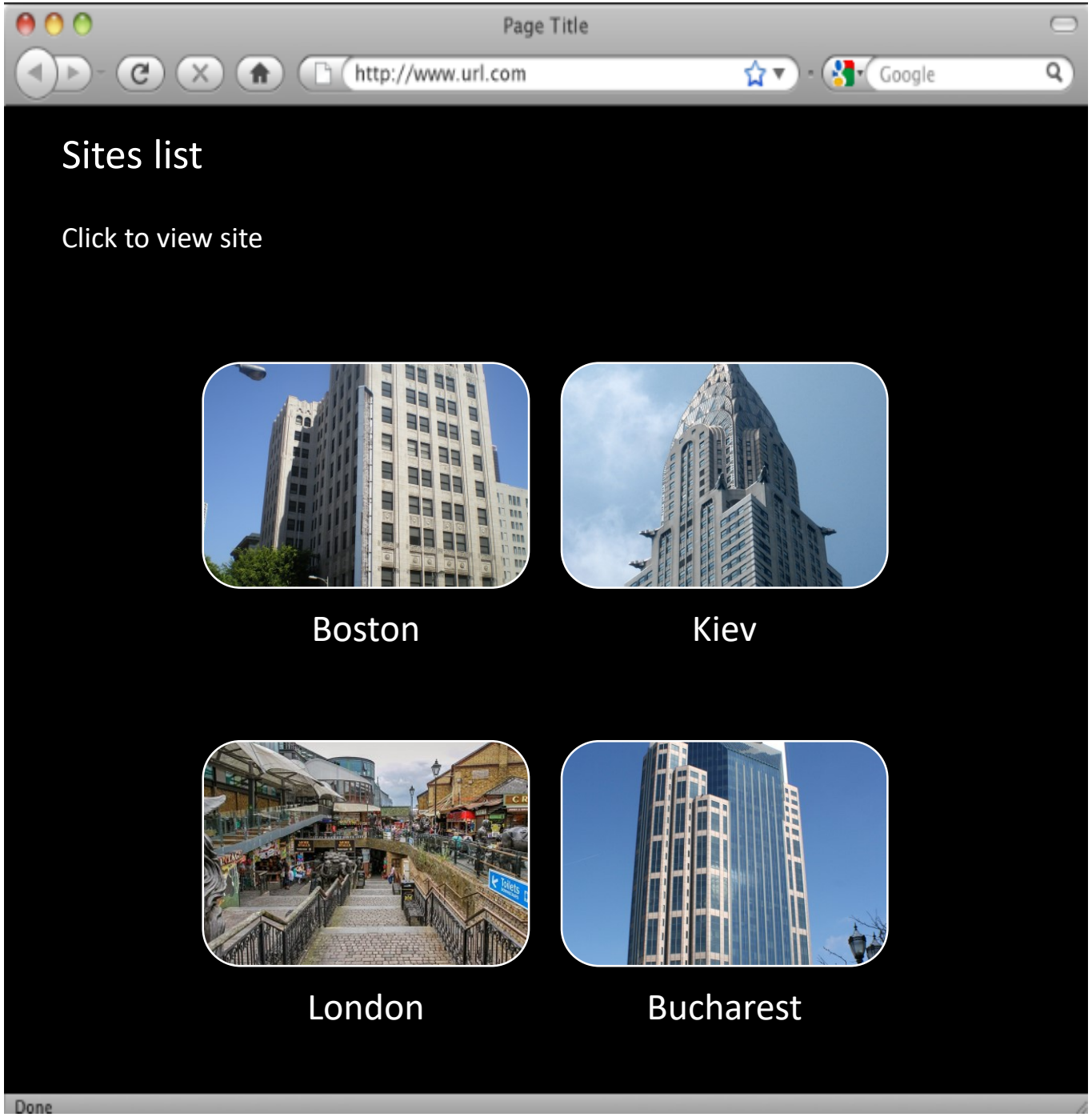
- Feel free to design according to your UI-UX perceptions.
- Responsive: PC 24" / Tablet 10" / mobile 5"

Languages and support

- RTL vs LTR:
  - Left to right (English)
  - Right to left (Hebrew, Arabic)
- Menus and alignment accordingly.

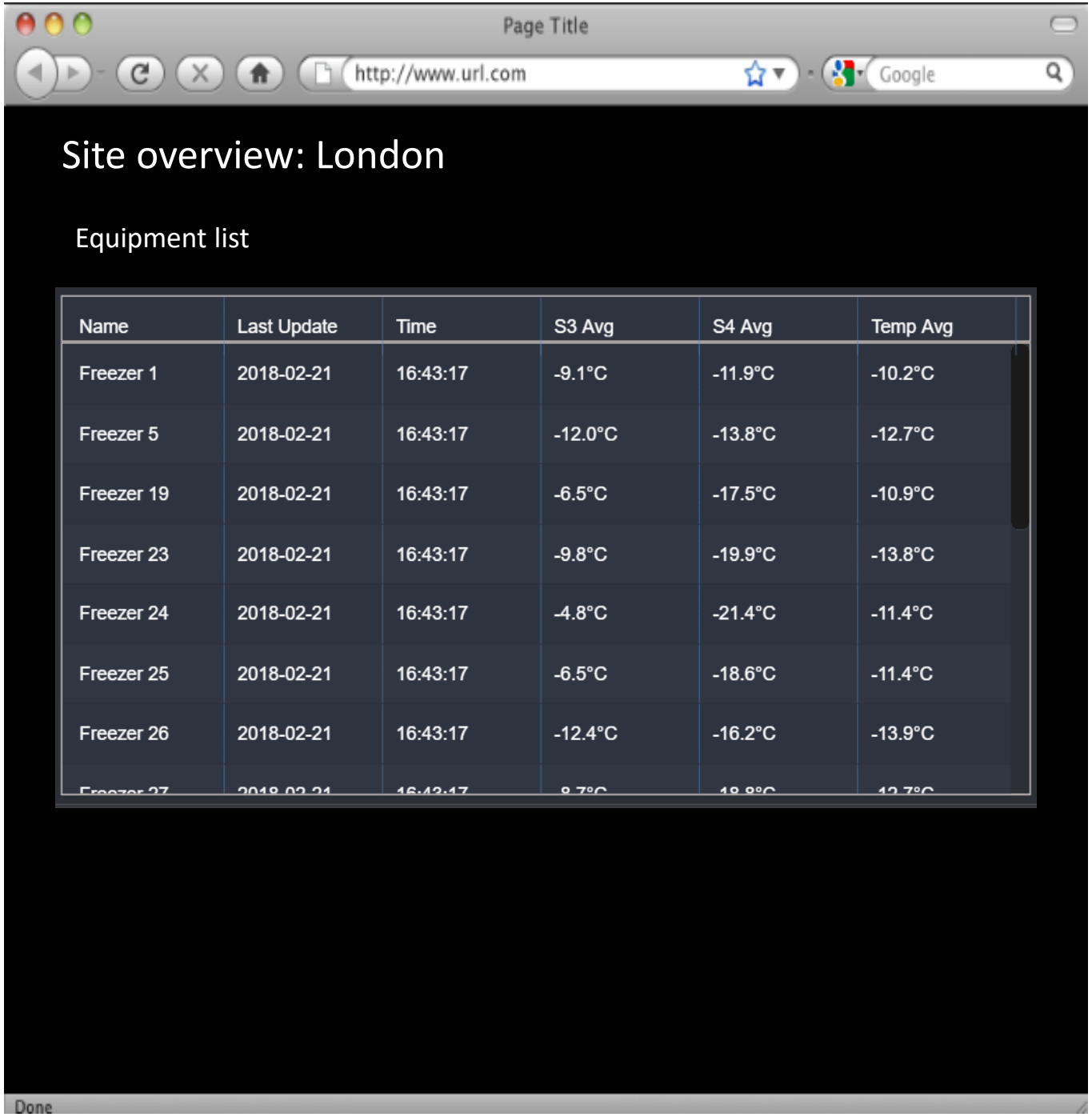
Sites list

- Click on site image or name links to next page with relevant site.
- layout to be responsive.
- List of sites from json



Site overview

- Site name is the one that was chosen in previous view. (in this example: London was clicked in previous view)
- Click on each row links to next page and shows the clicked equipment. In this view “Freezer 1” was clicked and was detailed in next slide’s view.
- Table to be responsive on mobile.
- List of equip and data for table from json
- Transform the json date format to yyyy-MM-dd (apply native date pipe)
- Could be lazy loaded and have a resolver service (as a bonus question).



Equipment analytics

- Site and equip names follows the chosen site and equip in previous views
- Graphs data is random in each time entering the view
- Each signal in different color
- Click on the checkboxes below graphs – hide/show related signal
- Graph data from json

Delivery requirements

- Delivery date: Sunday, March 11, at 20:00 PM
- Delivery content:
  - Link to your hosted app, with minified/unified code
  - Link to your github repository with app code
  - In github repo’s readme file - Instructions how to run unit-tests
- Delivery to: jobs@smartgreen.co.il