Senior Web Developer Task

The task is to build a Single Page Application (SAP) for showing data received from the client regarding tests results of trainees.

You can build using any library you want, with any tool you find over the WWW.

Important notes:

1) State of a page should be saved. For instance, if in a certain page I've used a certain filter, coming back to this page (without refreshing the screen) should show the same state as before.

2) Make sure everything works - filters, clicks etc. There's an automated script test that shouts about these errors...

3) In pages with filters - empty filters mean "Show All", and the filters are activated only when a user enters values.

Bonus points:

1) Using angular would rank higher

2) Adding unit tests to your app would add more points (you can add 2-3 to show your unit testing skills).

3) Documentation would be very much appreciated...

4) Graphic Design as you wish (most libraries come with their own design, so we did not want to limit you to a certain design for this test - you can use bootstrap, angular material design or any other design project you find suiting your needs). HINT: we LOVE angular material design...

Here are the apps snippets:

Data page:

Filter: Would filter the grid according to the text in the input box. The input can accept keywords like: "ID:" which will tell the filter to filter according to the ID column. It can also accept ">" and "<" for the Grade and Date filters to filter grades/dates in a certain range.

Grid: Would show the data in a table. Selecting a row would open the details in the details panel.

Details panel: Would show the data about the trainee. The data is editable and can be saved using the buttons above. The grid is limited to show 10 results on one page. Each row is a test result of a trainee in a certain subject.

Details panel buttons:

> *Add*: This button adds a new trainee to the system. It will open a blank details panel as well as add a new row to the grid.

> *Remove*: Active when there is a selected row, and clicking it removes the trainee from the system.

Pagination: Would enable to user to navigate between the rest of the data.

Analysis Page

IDs: Multiselect to select what ID's of trainees to present in the charts 1 and 2 (that show data per trainee).

Subjects: Multiselect to select what subjects to present in the chart 3 (that show data per subject).

Charts: The page can show 2 charts at a time. A user can drag and drop the charts to replace their location

Bottom hidden chart button: This button can be dragged and dropped in a chart placeholder and replace the chart. For instance, in the example above, dragging the "Chart 2: ..." to the left panel would show chart 2 in the left panel and Chart 1 would now be in the bottom hidden chart button.

Monitor Page

The monitor page shows the list of trainees and weather their average is above 65 or not. If it is, they are marked as Passed in green. If not, they are marked as not passed in red.

Filters: would allow to filter results according to ID's (multi select), names (free field) and state (checkboxes for failed and passed - default - both checked).

Grid: would show the data accordingly.

Good luck,

Yonatan