



Technical Test - FrontEnd

Subject

Reproduce the Phantoms dashboard of a PhantomBuster user

Functional description

This exercise aims to functionally reproduce the Phantoms dashboard of a PhantomBuster user, similarly visual but without necessarily reproducing it exactly.

The dashboard page (accessible only if you are logged in):

<https://phantombuster.com/phantoms>

Json payload of Phantoms to show as well as the corresponding TS interface are attached.

Page must contain:

- The Phantoms list with at least
 - Name of the Phantom
 - Drop-down menu with options “rename”, “duplicate” and “delete”
 - Options has to be functional
 - No persistence to change when page is refreshed
 - Launch frequency
 - Time remaining before the next auto-launch
- Categories filter
 - Show only the category name, no icon
 - Filter only one category at once
 - Filter persistence by URL when page is refreshed

Technical specifications

- Code in complete strict TypeScript (strict null checks, no any, ...)
- Only use <https://tailwindcss.com/> for all styles
- Use React with hooks
- Do not use redux or similar library
- Code at least one unitary test (example: remaining time display)
- Load the given payload like an external API call

Don't hesitate to ask questions in this email thread. What interests us is to see your method, your way of thinking, your use of React.

Visual aspect of final render does not need to be identical to the original counterpart, neither in shape or colors (but this is always better to have a minimum of estheticism)

Bonus points

- Make exercise classy and convenient
- Make exercise easy to visualize and to test
- Document it
- Add a path, links to a detailed view of each Phantom and a very basic view of the Phantom
- Add search functionality
- Have a countdown of the time remaining before the next auto-launch (`nextLaunchIn` is in seconds in the json)
- Drag & Drop to reorder Phantoms without persistence
- Use localStorage as a cache to make everything work persistently (the list of Phantoms as well as the "rename", "duplicate" and "delete" actions) If datas are cached, ignore the API, add a button to reset the cache and reload the API.
- Any other things that you want to or for trying to impress us more 😊

Needed files

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/65beaa65-b41b-4b86-94b8-1ddc6c479ed5/phantoms.d.ts>

The link is for intern purposes, please use attached file

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/6d3385a7-8b87-4de3-9090-6366e8fa7c68/phantoms.json>

The link is for intern purposes, please use attached file

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/c6f7c37b-ce2f-4a8f-a475-a5720a5316a5/phantoms.zip>

The link is for intern purposes, please use attached file