**Overview - Conceptualizing the product**

We would like to see a UI which at a minimum can:  
  
\* Interface with and retrieve a JWT token from the login endpoint  
\* Display and allow setting of the user’s profile image  
\* Retrieve a list of countries and their dialling codes for use in submitting contacts  
\* Display any error responses received from the backend to the user.  
  
\* 500 codes should be presented with a suitable text to inform the user to contact support if the problem persists  
\* 400 codes should be handled similarly.  
\* The non-standard use of a 418 error code indicates that the error message is to be displayed to the user.  
  
\* Allow the user to change their password  
\* Allow the user to save, update and delete contact details  
The API documentation can be found as interview.intrinsiccloud.net/swagger-ui/  
Username: user3@intrinsicgrouplimited.com  
Password: Pa$$word

**Site map**

**Page templates: design**

1. Heavy use of Material UI. This is something you will find people who build multiple small generic websites use as an alternative to bootstrap. That is OK when you are starting out, it’s an easy way to build screen quickly however we really need people who understand building components themselves.
2. 4. UX experience. This is a chance to showcase some of your ideas for a commercial user facing website, we just found your design a little unappealing. We don’t want multi colours flashing fireworks bit square gey boxes are a bit utilitarian

Login, not grey, rounded edges, error message in some coloured text

Dashboard, bit of a menu including a logout as my login now persists

One container that would show profile pic, with the ability to change, nicely formatted details with headings, then the current & new password with a change password button.  
\* Second container, list of contacts, be good if this had a freetext search option so I could type part of a name, country or number to filter the contacts. Scroll should be inside this container so the screen remains static, and the contents scroll. Only show the name maybe and when I click on a name fill the…..  
\* Third container. Shows contact details when clicked in middle container. Empty when screen first shows, empty of I type in the search box to change the filter.  
\* Data in boxes like the attached maybe with headings.  
\* Buttons  
\* UPDATE – this should show if there was a contact selected and any of the details changed.  
\* ADD – this should show either if the contact was select and the name has changed or the contact details were empty and a new contact details were added.  
\* DELETE – This should only be shown if a details are filled and the contact exists and is up to date. Should not be shown if the UPDATE or ADD buttons are shown.  
\* CANCEL – show if any of the detail data has been hanged. Should replace DELETE button. Should revert and existing contact to pre changed state and clear the details if no contact was selected.  
Footer, put a little footer at the bottom of the page with your name on or something like ‘© Copyright 2022 – Viktor Hulei’ show you are happy to b associated with it.  
  
Lastly, make it responsive, the header and the footer don’t have enough in them to be a problem if you shrink the page to mobile size but the 3 boxes on the dash should shrink to the point they can then jump to a column of 3 boxes instead of a row. Look at flexbox for that.

* desktop and mobile
* hover
* animations

**Page templates: functionality**

. Also it’s a lot of clicking to do things, there is enough screen space to put 2 or 3 nicely formatted columns one of which could have a scrollable container of contacts maybe, this could have had a flexbox parent so the same webpage would display in columns on a mobile device. The photo upload still doesn’t work for me BTW

error message in some coloured text not a big in my face box

**Development schema**

State management/store. I can see you have a very basic implementation of ContextAPI, use of this needs to be much more indepth to show a full undertstanding of state management. We, as most other commercial organisation, use Redux . Try a look at actions and reduce, a full understanding of these will be crucial in a commercial environment.  
2. Axios, you have all your api calls in the comlponent that uses it. These should really be in their own folder and separate components; these could also be used in conjunction with reducers . Maybe try to look at these AxiosRequestConfig, AxiosResponse and Axios Interceptors.

When you log in get all the profile data and all the contact data and put it in a store. You could update this data when changes are made and send the POST in separately in the background but you’ll have to be careful to roll back changes if the POST fails.  
  
Updating the profile picture should take effect immediately not after the save, same principle as above. No loader but if the POST fails go back to the old image and show a neat message.

1. Authorization
   * errors: wrong credentials, network failure, problem with server etc.
   * success
2. Loading and displaying:
   * profile data
   * contacts data
3. Changing photo functionality
4. Changing password functionality
5. Adding new contact
   * buttons appearance
   * add button
   * cancel button
6. Deleting a contact
   * buttons appearance
   * delete button
7. Search and filter
8. Updating the contact
   * buttons appearance
   * extracting and adding phone numbers
   * update button
   * cancel button
9. Error messages

**Folder structure**

Gfgf

Gfg

**Component testing**

Gfgf

**Browser and device support**

Gfgf

gfgf