Story UFC-187: Account Deletion

The following document will go over the design of account deletion, as outlined in Story ticket UFC-187.

Acceptance Criteria

For reference, the following are the acceptance criteria for this ticket:

- I can click on a button to delete an account.

- I am prompted to confirm that I want to delete the account.

- I can undo the deletion of the account for a short period after deleting it.

Design Brainstorming

ModalDialog

Note: The `ModalDialog` component got renamed to `ConfirmationDialog` because it made more sense that this was a higher level, opinionated component than a generic modal dialog. However, I'm lazy, and I didn't bother updating all references to it. So deal with it!

To start, we only need an error variant. Then again, I don't really see the need for *other* types of modal dialogs (when would we have a success modal? Maybe a neutral confirmation modal?), and we definitely don't have the need right now.

In terms of implementation, I think we can get away with making the base ModalDialog quite opinionated: the title, icon, close button, and the two actions (primary + cancel) can all be part of the base layout.

Then, the ModalDialog component can take children that it renders in its body.

In terms of props:

- variant: string = "negative"

- title: string

- primaryActionLabel: string

- secondaryActionLabel: string = "Cancel"

- onClose: () → void

- onPrimaryAction: () → void

- children: React.ReactNode

In terms of how to control the modal visibility, I was originally planning on converting them to the URL based visibility that the AccountForm uses, but I'm having second thoughts.

First, I took a look again at the sagas and realized that they control the entirety of the modals' lifecycle -- showing it and then waiting for cancellation/confirmation. I feel like rewriting them so that they can support the URL based visibility would be a non-trivial amount of work.

Second, I don't think users need URL based access to the modals nearly as much as something as the AccountForm.

As such, I think we should just keep the current paradigm of visibility controlled by the store.

The only extra thing we'll need to do is modify the BackgroundBlur to support visibility from the modal dialog visibility values, on top of route based visibility for the AccountForm.

The only support annoying thing about this is that we used 'modal' a lot when naming things for supporting displaying the AccountForm from a URL -- and not the actual 'modals' aren't even using them. So, awkward...

Component Breakdown

Atoms

- CircularCountdown

- [modification] ShadowButton (add red 'negative' variant)

- [modification] BackgroundBlur (add visibility control by modal store visibility)

Molecules

- LargeToast (IconButton, TextField, Icon, CircularCountdown)

  - With Success, Warning, and Error variants.

- SmallToast (IconButton, TextField, CircularCountdown)

  - With optional action (i.e. Undo)

- ConfirmationDialog (IconButton, negative ShadowButton, LinkButton)

  - Start with just a 'negative' variant.

Organisms

- AccountDeletionDialog (ConfirmationDialog + body)

Scenes

- ToastMessages

Tasks

- Build the toasts.

- Build the modal.

- Connect the Delete button in the Accounts List.