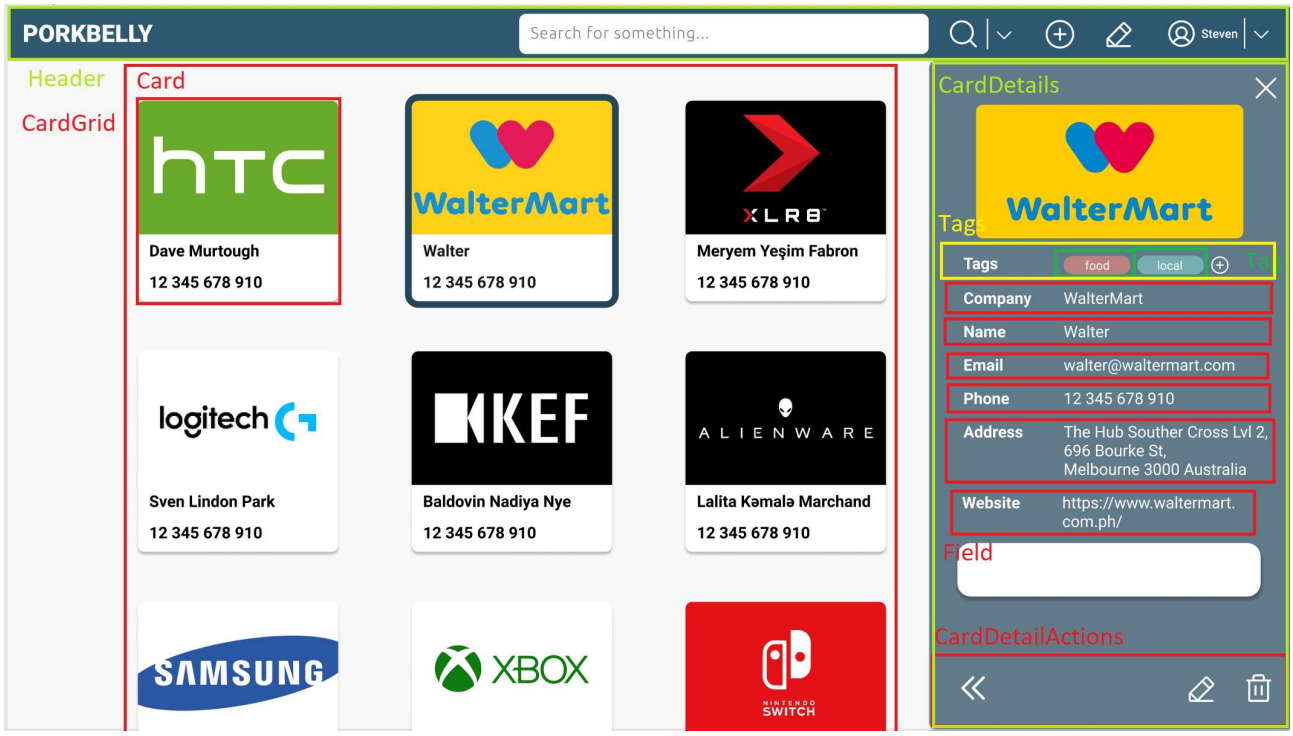
Team PorkBelly
Meeting minutes
30th August, 2021

Composed by Shang Zhe in attendance of Richard, Yujian and Walter, Parit (arriving halfway)
Meeting begins at 2:09pm

Shang Zhe began with the home page, starting from the smaller components. Card components were identified. These would be organised into a CardGrid. Within the right hand slid in div, card details would organised as Fields that capture the card's keys and values. Card Tag were also identified, arranged into the Tags.

Button controls CardDetailActions were grouped together at the bottom of the panel, aptly named CardDetails.

The website's Header was identified. Its components were identified further.



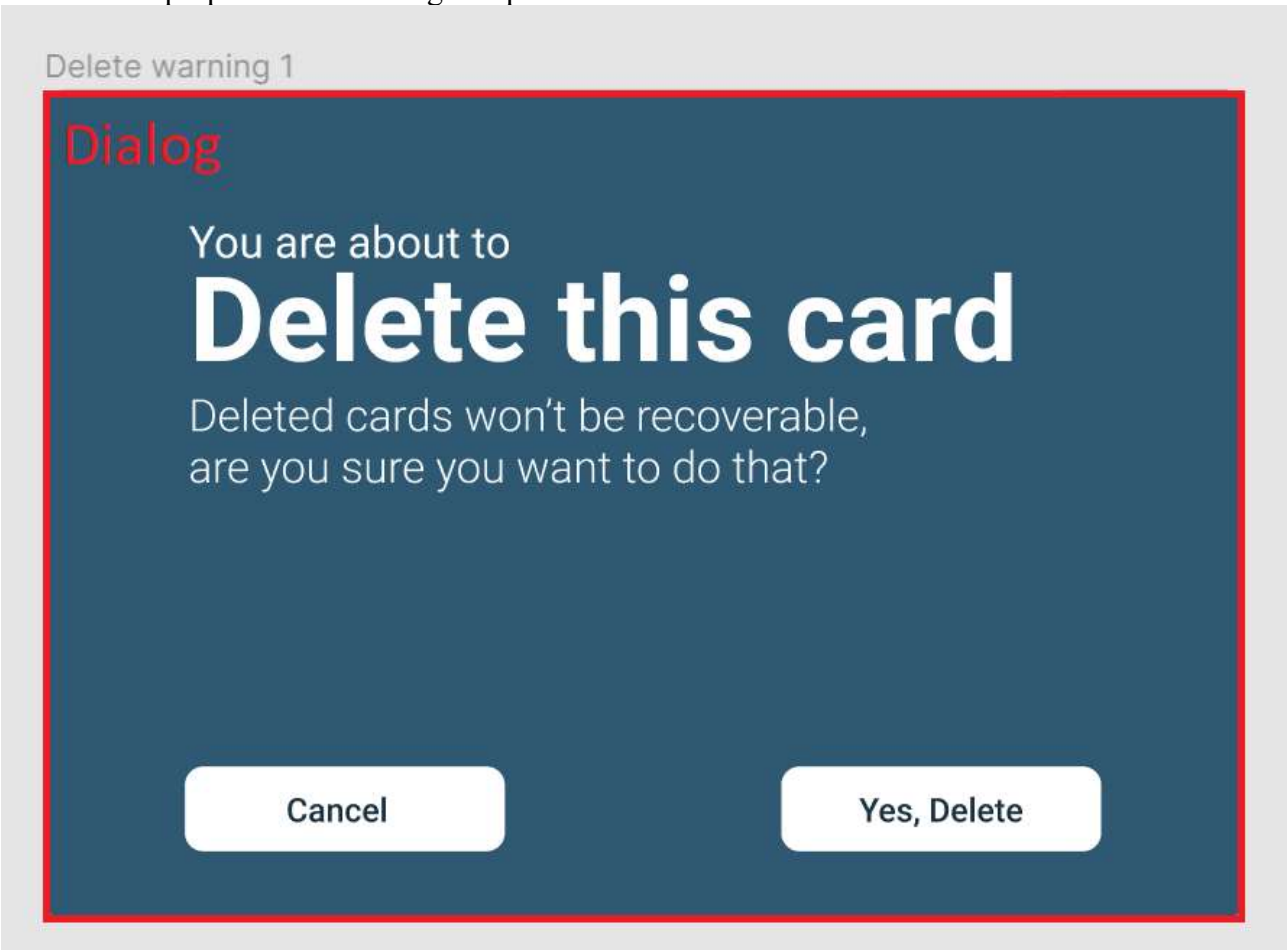
On the topic of header components, Shang Zhe suggested the CommandBar from FluentUI. It might create a new row beneath the header, which Yujian and Shang Zhe found wasteful of the space on the right. Yujian saied that the CommandBar might not be on a second bar, instead either on the header itself or floating.

The next component broken down was the SearchBox. Shang Zhe and Yujian looked at the components in FluentUI for their search boxes, hoping to look for an auto-complete feature. Richard observed that the search option could be implemented as a keyword search. The search could be resumed in the future – potentially in the next iteration.

Yujian described that user feedback from the previous prototypes requested the SearchBar's icons be placed inside the bar itself, rather than outside. Shang Zhe asks on the importance of the search button drop down. The dropdown was found to be related to Sprint 2's Field search.

Shang Zhe then enquired on the need for profile dropdown. Yujian identified it as a gateway to log out, user edit, and potentially a dark ode (which Shang Zhe stated that it could be done with ThemeProviders). The profile dropdown Shang Zhe claimed could be an atomic component.

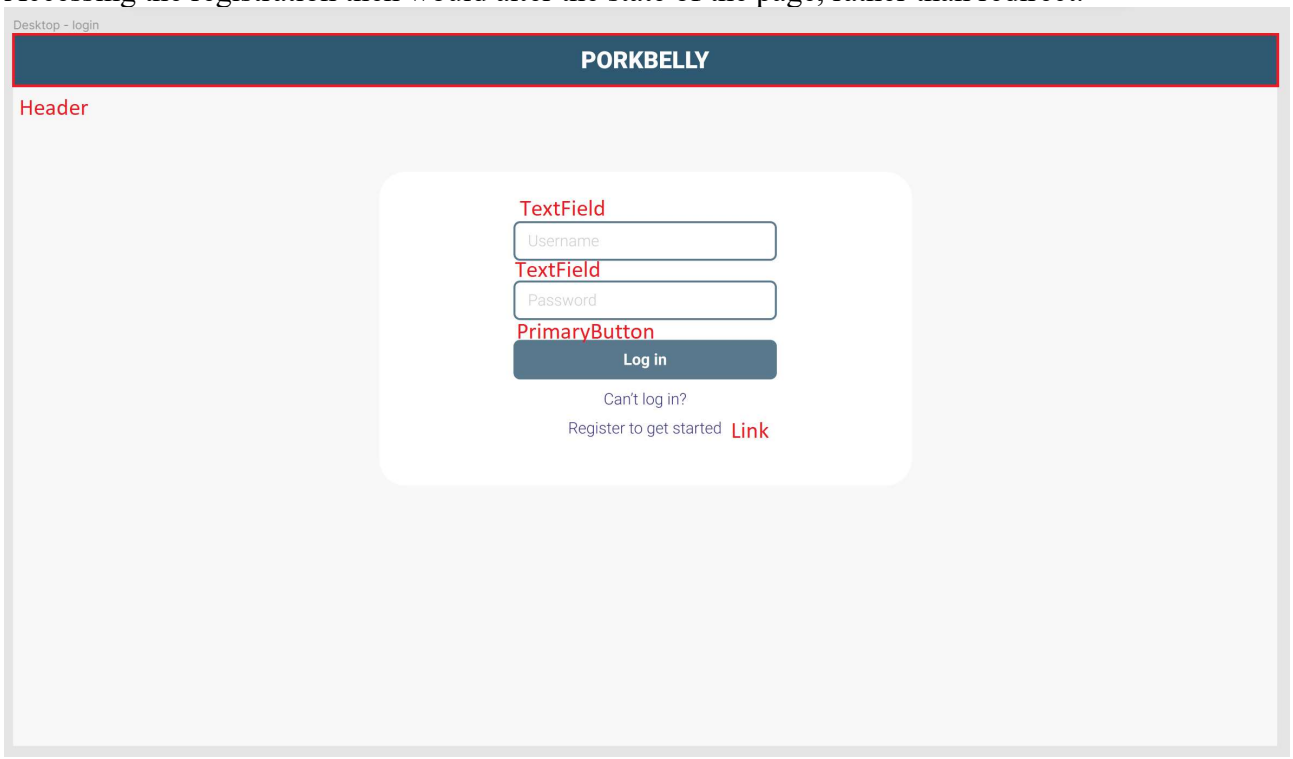
The breakdown process then moved to the DeleteWarning. This component, Shang Zhe said, matched the purpose of the Dialog component from FluentUI.



Discussion fell upon the add card button inside the CardGrid. Shang Zhe. Other than requiring to create an asset for its symbol, it required different treatment from other Card components. With a lack of unique functionality – as Shang Zhe pointed out the add card control in the header – Yujian decided that it was acceptable to omit that component.

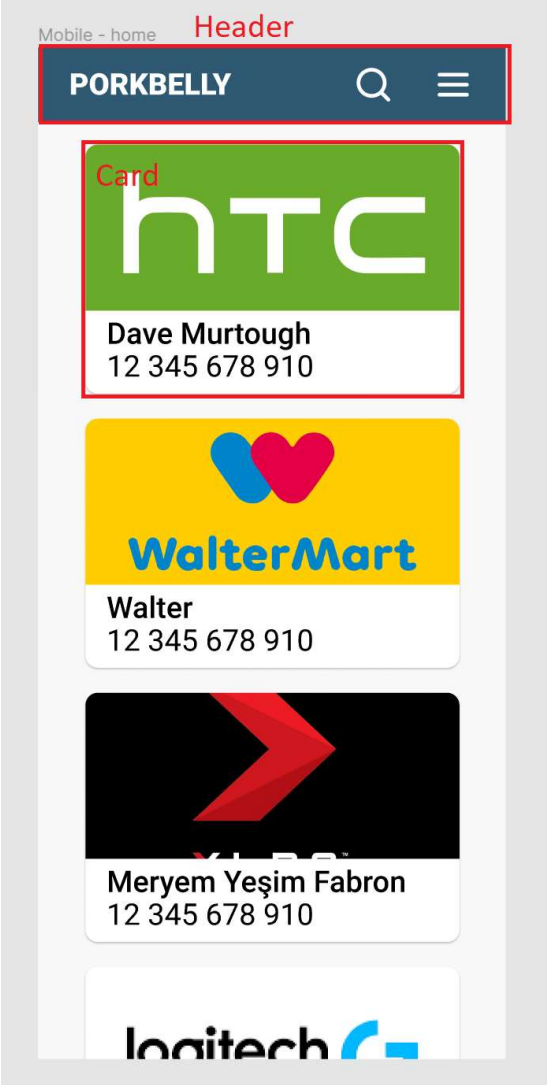
While examining FluentUI's CheckBox component for a batch card, Shang Zhe recommended replacing the pencil icon with a checkbox. It was decided that batch selection would be deferred to a later time.

For the login page, the Header was to be recycled with state to alter its appearance. TextFields were identified and the PrimaryButton were confirmed to act as a log in and registration controller. Accessing the registration then would alter the state of the page, rather than redirect.

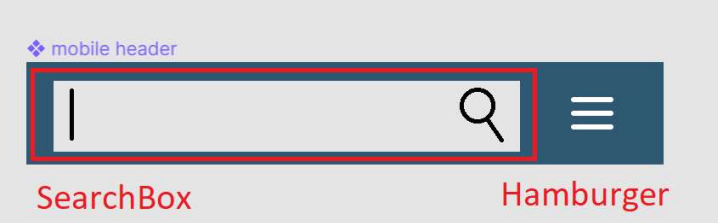


<Parit joined at 2:42pm – having just received his second vaccination.>

The Mobile view was analysed next. A majority of the components could be ported over.



Shang Zhe recommended that the search feature on mobile would not take up the whole screen, rather becoming embedded within the header. The Hamburger would be modelled as a button that reveals a Panel component with the burger's options.



Minor design tweaks were that reflected the change of search bar and panel menu. Included with the header functionalities, card fields on the mobile view were updated to house their values beneath the field keys, in order to maximize field width. Tags in the mobile view were also to be provided with a full column's width, and to grow vertically.

Yujian would take this into account finalising the UI design.

Walter suggested that the application logo remain in the hamburger menu – a case of branding. Yujian felt unsure on the aesthetics. Richard requested having icons in the menu to spice up the

overlay menu, and Walter added to that saying that both icons and text should be included so it would be span the width of the menu.

Shang Zhe called that during the workshop the team could determine the prop types for each component, or as soon as possible.

The meeting is adjourned at 3:07pm.
https://www.figma.com/file/eTM4bPj4803ZOjXDyxsmOk/UX-prototype-v2?node-id=0%3A1

Meeting resumes at 4:47pm.

[Yujian takes over minute taking]
Richard suggested that we needed a new issue to support card notes. Shang Zhe insisted that an attribute should be fine. A little back and forth, no conclusion.

Meeting became an exercise to define the properties for each of the broken down components Shang Zhe was leading the task. The information were represented using TS. The objective was to choose the props for each components. Also define the controllers, so that it can be mocked during development, and that we can individually test the components.

Cards.tsx, the cards would just be an array.

Cardgrid.tsx would only manage the layout of the cards.

Shang Zhe drew the layout using microsoft paint.

Discussion moved onto card detail fields.

CardField.tsx, each card field would only have a single field, no need for array.

[Walter returns from a break to take minutes]
Parit asks the behaviour of the cards schema of a tags deletion.

Walter asks about the naming conventions for these react props and interfaces, and how the interfaces would be captured in the documentation. Shang Zhe responded saying that these interactions and property types could be captured in the model.

Shang Zhe created controller folder, and subset of files, that will contain methods to house api calls.
Life cycle: eg card grid

1. Server would render card grid without card, while the api call needed to be waited.
2. Components would need to call methods in a controller that would do the api call and set the state of the components using the result of the call. Controller interfaces were defined in the meeting, for implementing classes to realise them.
3. These controllers would handle api calls and would mostly be used to affect the Context values in the front end, which might propagate changes to child components, even overwriting their state (could be done with a combination of useState and useEffect).

Walter asks whether these interfaces would like to capture the states of these Components. Shang Zhe is uncertain whether changing the props to a component will require state hooks to be reinitialised.

Discussion led to confusion about the commit() functions within the controller interfaces. React Context can be used to signal all the implementing consumers to update: List of cards, list of tags, user contexts could be used in Context. React State could be used for smaller component properties such as EditingState for the CardDetails panel.

Walter asked how states and props could be mingled, how to determine whether props or states should be used. Shang Zhe stated that props could be used to define a Components starting State. This discussion was not related to the controller interfaces.

Shang Zhe told the attendees that he would be fine carrying in preparation for size estimation.

Meeting ended at 5:54pm.