TELUS Component Library

Responsiveness

Multi-Platform Component

Card 1 is here to display the current viewport size: xl

Card 2 is on the right

```
function useResponsivePropExample() {
  const viewport = useViewport()
  const secondCardPosition = useResponsiveProp({ xs: 'in the middle', md: 'on the right' })
  const thirdCardPosition = useResponsiveProp({ xs: 'at the bottom', md: null })

  return (
    <StackView
      direction={
        // Note that some components (e.g. `direction` in `StackView`) have props that take responsive objects
        // (e.g. `direction` in `StackView`) and use `useResponsiveProp` under the hood:
        { xs: 'column', md: 'row' }
      }
    >
      <Card>{`Card 1 is here to display the current viewport size: ${viewport}`}</Card>
      <Card>{`Card 2 is ${secondCardPosition}`}</Card>
      {thirdCardPosition && <Card>{`Card 3 is ${thirdCardPosition}`}</Card>}
    </StackView>
  )
}
```

Follow the appropriate instructions to add this component in to your app.

Guidance

useViewport

`useViewport` hooks helps you to get the currently used viewport. You don't have to pass any parameters, and it returns the string matching the name of the viewport.

useResponsiveProp

You can use `useResponsiveProp` hook to create values that change depending on the current viewport. This hook takes any value and, if that value is a responsive prop (an object with one or more keys matching system viewport names), returns the value corresponding to the largest viewport key smaller than the current screen's viewport.

For example, if passed `{ sm: 4, lg: 8 }`, it will return 8 on `xl` and `lg` viewports, 4 on `md` and `sm` viewports, and `undefined` (or a provided `defaultValue`) on `xs` viewports.

> INFO
>
> Note that `useResponsiveProp` uses `useViewport` under the hood, so you should go with `useResponsiveProp` by default and leave `useViewport` for cases with more complex logic.

Responsive component

You can also use the Responsive component if you prefer it to the hooks.

Platform considerations

The component is available on both native platforms and web.

Feedback

- Spotted a problem with this component? Raise an issue on GitHub
- See any existing issues for this component

- Contact the team on slack in #ds-support