

This project was bootstrapped with [Create React App](#).

Executing the application

```
npm start
```

Development process

This project is bootstrapped with Create React App and relies on it for all build nuances.

Application consists of 2 main components - `InfiniteScroll` and `Message`.

InfiniteScroll

`InfiniteScroll` is responsible for fetching data from API, rendering of the list of messages and vertical scroll functionality.

When mounted component measures available view size and determines how many messages are visible at the same time (based on the predefined minimum message height since messages vary in actual content height) - this is the number of messages that are going to be rendered at any given moment. This calculation is updated on window resize.

Then component fetches messages from the API endpoint as well as attaches event listeners required for the scroll functionality.

On component update check is performed to make sure that there's enough messages left at the end of the queue - otherwise another fetch is initiated.

Component has a `deleteChild` method that removes specific message from the component state and `hideChild` which is triggered by the `Message` component whenever that specific message disappears from screen. `hideChild` expects message height as a parameter that is used to increase list padding offset to prevent sibling shift when message is removed from dom - these height values are stored in state and treated as a stack - when users scrolls in the opposite direction reverse process takes place.

The scroll logic is controlled by `onMove` method that separates vertical scroll from horizontal drag action on the message (in which case messages are notified that user is attempting to side swipe and enables message drag functionality). In case of the actual scroll, this method is responsible for maintaining correct visible bound on the messages.

Finally, `render` method applies position change to the list through `translate3d` in order to utilize hardware acceleration.

Message

`Message` component represents individual messages and is responsible for message removal gesture as well as detection of when the message has been scrolled off the view.

This component has attaches drag related event listeners to it's own DOM container and performs message horizontal shift when parent component is not scrolling vertically. When message is moved beyond specified threshold, parent is notified that message needs to be removed.

What would you do differently if you had more time?

- Modify `Message` to add event listeners only when the component is in `isDraggable` mode.
- Add smoother animation when removing the messages
- Introduce kinetic scrolling effect (to continue scrolling after release depending on the speed/distance of the swipe)
- Revisit list offset padding approach as i am not entirely happy with it
- Simplify `onMove` method