CSE183 Fall 2021 - Assignment 5

In this assignment you will write a React and Material-UI Web App based on wire-frame designs given to you by a supposed client.

This assignment is worth 7.5% of your final grade.

Installation

See instructions in Assignment 2 and ensure you are running the current LTS version of Node.js.

Setup

Download the starter code archive from Canvas and expand into an empty folder. I recommend, if you have not already done so, creating a folder for the class and individual folders beneath that for each assignment.

The starter code archive contains one file you will modify:

```
src/App.js
```

And some files and folders you should ***not*** modify:

```
public
src/__tests__
src/data
src/index.js
package.json
```

Adding JavaScript / JSX files for new components to the `src` folder is a good idea, but do not create sub folders and do not add any additional tests.

To setup the development environment, ***navigate to the folder where you extracted the starter code*** and run the following command:

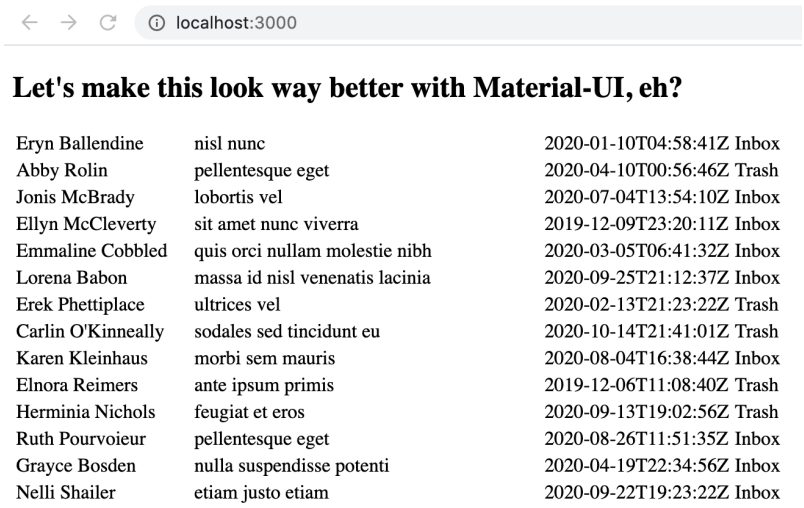
```
$ npm install
```

This will take some time to execute as `npm` downloads and installs all the packages required to build the assignment.

To start the dev server, run the following command:

```
$ npm start
```

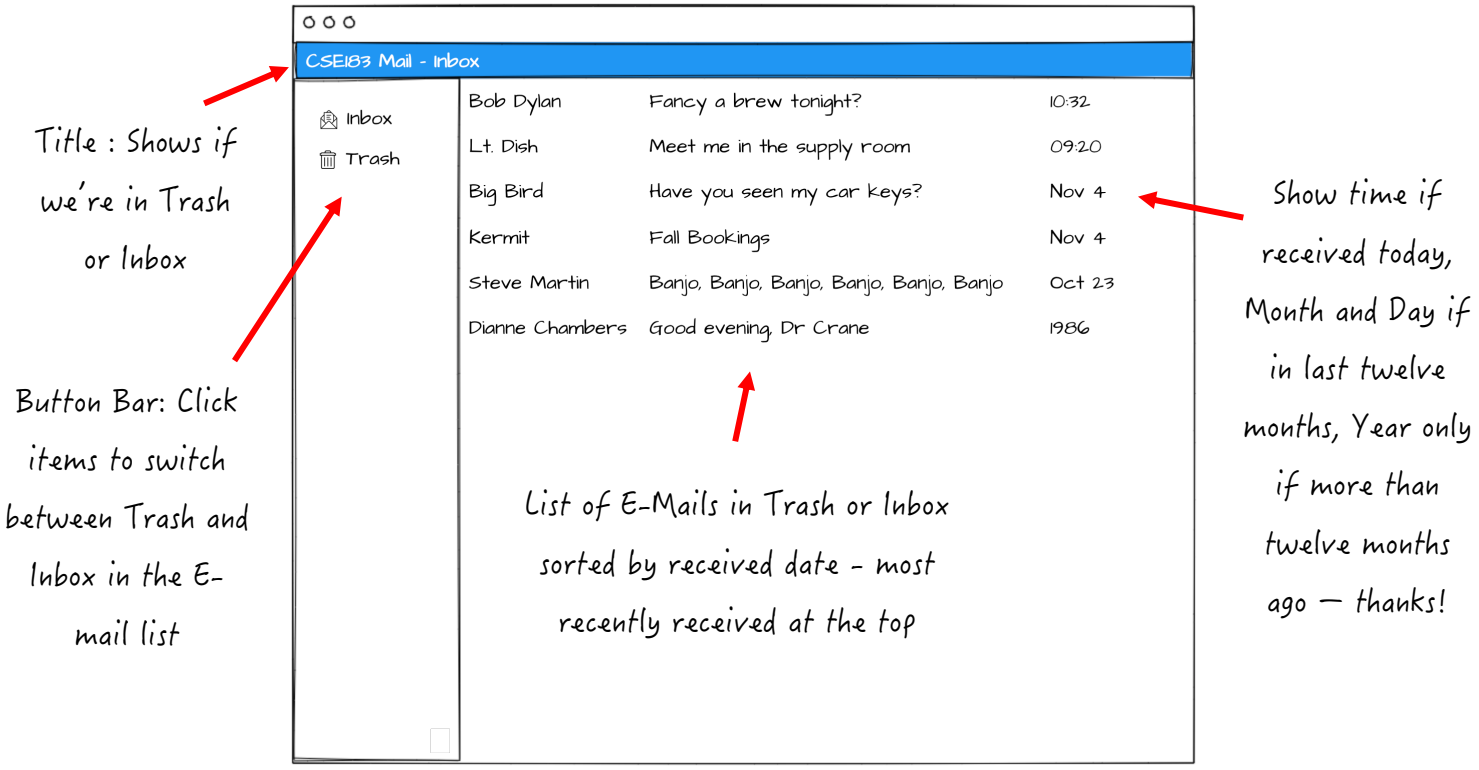
A new browser window will open:



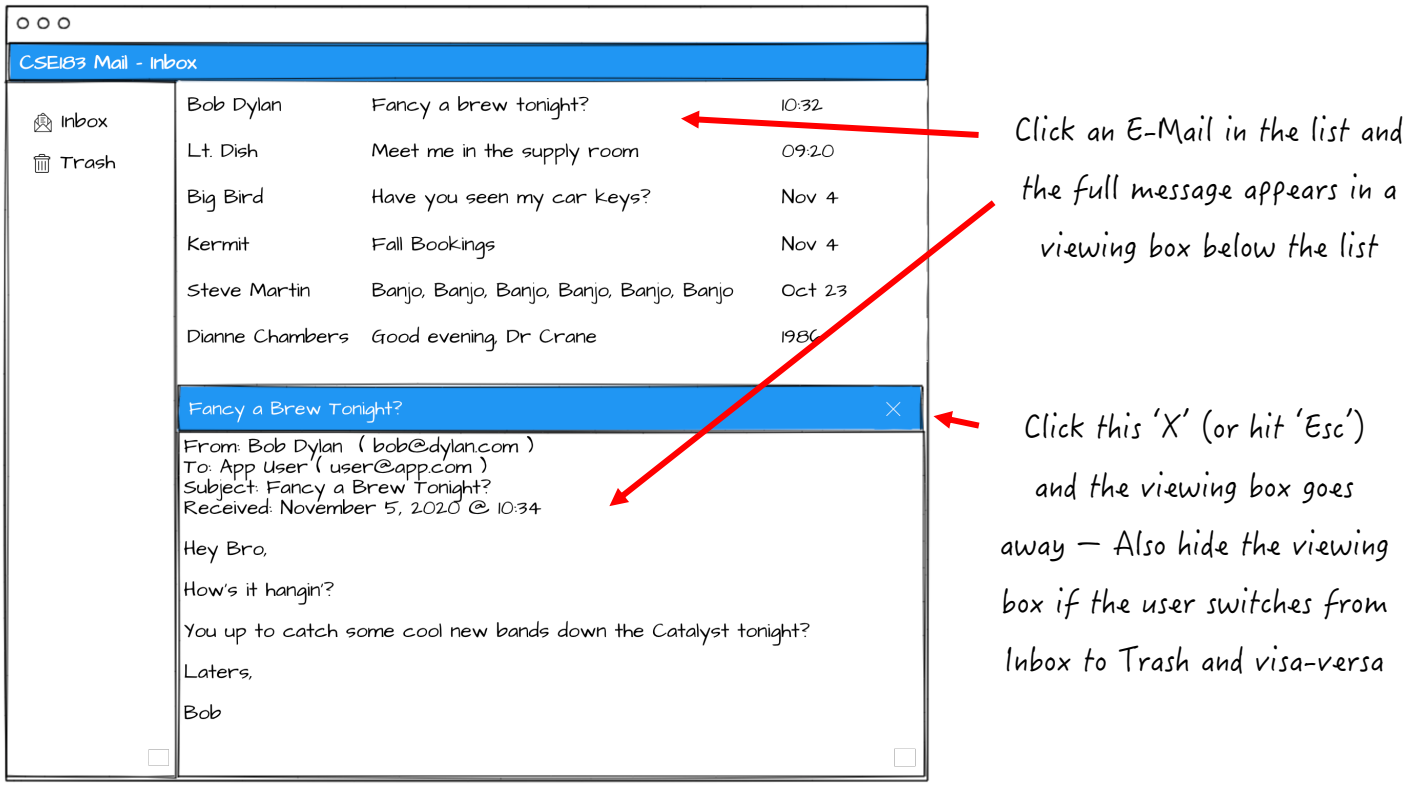
Requirements

A client has given you a set of wireframes / mock-ups for an e-mail reader they want you implement in React and Material-UI. Your client has a number of requirements…

Desktop 1:

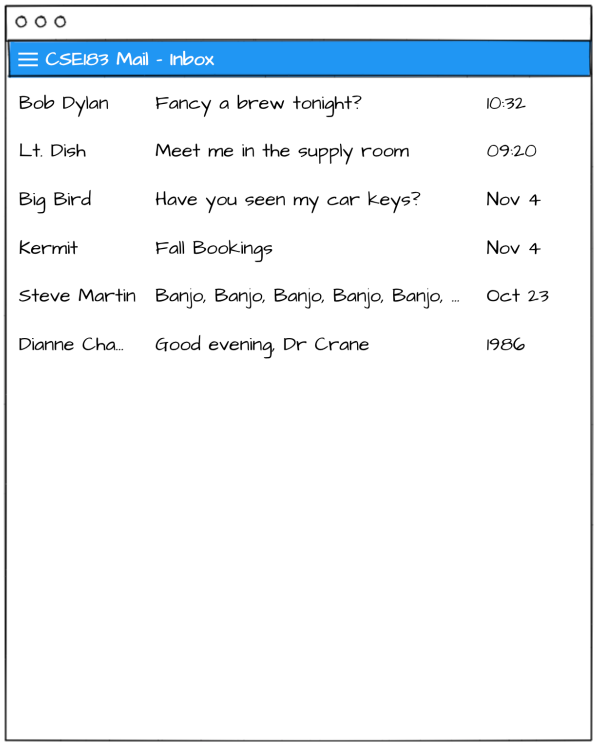


Desktop 2:

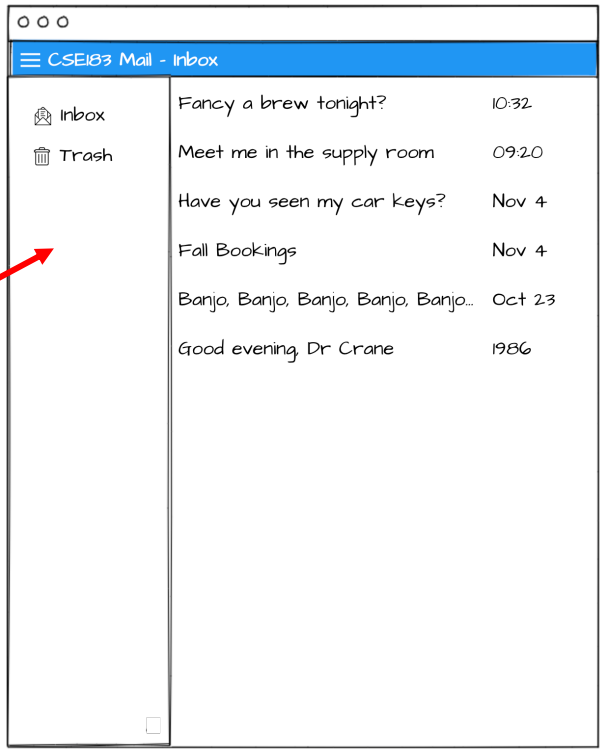


Desktop 3:

When the window is reduced in width to the point where the message titles are getting too short, hide the Button Bar and replace it with a Menu Icon to the left of the title



If the menu Icon is clicked, show the Button Bar over the top of the E-Mail List. Clicking on the list or Title or Menu Icon should hide the Button Bar again. Clicking a Button Bar item should select the appropriate list AND close the button bar ☺

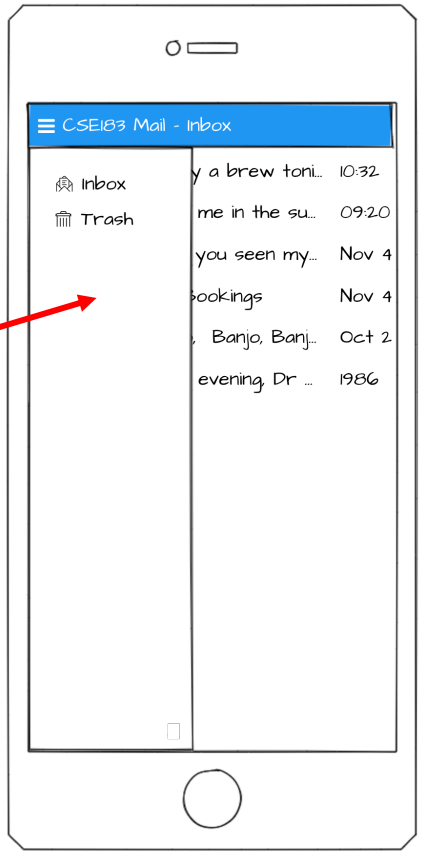


Mobile 1:

Show a menu icon next to the Title

Show the Trash or Inbox E-Mails using the full width of the phone screen

If the menu Icon is clicked, show the Button Bar over the top of the E-Mail List. Hitting 'Esc' and clicking on the list or Title or Menu Icon should hide the Button Bar again. Clicking a Button Bar item should select the appropriate list AND close the button bar – woo-hoo!



Mobile 2:

Click an E-Mail in the list and the full message appears and takes over the whole screen with the subject in the title bar

Click this 'X' (or hit 'Esc') and the E-Mail Viewer goes away and the E-Mail List is shown again

What steps should I take to tackle this?

Study the examples at https://material-ui.com/, play with their code in the provided sandboxes, then use some of it as starter code for your Web App. Don't be alarmed if you end up throwing a few attempts away, that's perfectly normal and a useful experience.

You'll probably find you don't understand all the code you use, but that's ok. Get it running then modify so it does what you want. Don't struggle for too long with something you can't get to work. Find another example and try that.

Some guidelines:

- Write functional React components with Hooks, not ECMAScript Classes.
- Avoid using Material UI `<Hidden>` components - they are deprecated in the next version
- Start simple and work up from there, a reasonable initial approach might be:
    - Create a simple email-list component
        - Initially show all emails
        - Ensure list renders well on desktop and mobile
    - Introduce state for which mailbox to view
        - Set the state in code
        - Check list is filtered correctly to only show Inbox or Trash
    - Add a permanent drawer with buttons to select which mailbox to view
        - You'll need a `Context` to share state between components

Lint-as-you-type

You'll notice that the assignment has been configured so your React Web App will not run if your code fails any of the linter checks. Your code will therefore need to be super clean and tidy before you can manually test it ☺

Much of the code you find on-line will fall foul of the linter checks - so fix it!

Testing

The suplied tests use React Testing Library https://testing-library.com/docs/ and the matchers provided by Jest DOM https://github.com/testing-library/jest-dom#usage. You do not need to write any of your own tests, but you may find it useful to study the supplied tests as you will need to write your own for your group project.

To run the supplied tests continually as you develop:

```
$ npm test
```

To execute all tests once, run the following command:

```
$ npm test -- --watchAll=false
```

To run the linter outside of the React development server:

```
$ npm run lint
```

How much code will I need to write?

Potentially very little. Use the examples at https://material-ui.com/ and anywhere else you can find code that helps you meet the requirements. Remember to give credit to the original authors.

Grading scheme

The following aspects will be assessed:

1. (60%) **Does it work?**

    a. Desktop Tests (40%)
    b. Mobile Tests (20%)

2. (40%) **Is it well written?**

    a. No linter errors or warnings (10%)
    b. Code Coverage (30%)
        100% (30%)
        >= 98% (20%)
        >= 95% (10%)
        < 95% (0%)

3. (-100%) **Did you give credit where credit is due?**

    a. Your submission is found to contain code segments copied from on-line resources and you failed to give clear and unambiguous credit to the original author(s) in your source code (-100%). You will also be subject to the university academic misconduct procedure as stated in the class academic integrity policy.

    b. Your submission is determined to be a copy of a past or present student's submission (-100%)

    ~~c. Your submission is found to contain code segments copied from on-line resources that you did give a clear an unambiguous credit to in your source code, but the copied code constitutes too significant a percentage of your submission:~~

        - ~~< 25% copied code No deduction~~
        - ~~25% to 50% copied code (50%)~~
        - ~~> 50% copied code (-100%)~~

    In this assignment you are encouraged to use as much code as you can find on-line - but remember to give credit to the original authors.

What to submit

Run the following command to create the submission archive:

```
$ npm run zip
```

**** UPLOAD** `CSE183.Assignment5.Submission.zip` **TO THE CANVAS ASSIGNMENT AND SUBMIT ****

