Submission 03 - Team Okra

Profile Page - Ryan Minichiello

The mount point appears on line 14 in */build/profile_temp.html* as <...id="profile_temp"> and the corresponding */app/components/profile_page/profile_temp.js* serves as the top level React component, calling on the subcomponents it imports and rendering appropriately.

Known Issues (more noted where I know the problem resides below)
- Most of the buttons are not dynamic or responsive yet, besides the message button in the banner. This will be addressed once we put together a dynamic page.
- The page breaks when squeezed horizontally, it seems to be a problem across all the pages of our web application. Using CSS to handle resizing should fix the problem, and a fix for one page will be a fix for all pages.
- The page is a bit too long overall, will be addressed through better modularization of the overall page components.

Component List
- Profile_Temp: Serves as the top level component, calling Profile_Nav, Profile_Sidebar, Profile_Banner, Profile_About, Profile_Skills, and Profile_Teams to render their respective components. It also outlines the framing divs within which each component will reside. This is the point at which all the data flows through and is properly filtered down.
- Profile_Nav: Renders the navbar that appears on the top of every OSTF web page.
- Profile_Sidebar: Renders the sidebar with corresponding list of current projects and a link to the job board.
- Profile_Banner: Renders the user's unique banner image, user image, and handles the messaging functionality by popping up a dialogue box when clicked.
    - **KNOWN BUG**: The banner image fails to fill the parent <div>, and attempts to fix by removing inline styling and moving to CSS file have resulted in the same problem as the inline, namely, setting width and height to 100% blows the entire div up and wrecks the intended design of the page. Plan to resolve by breaking this Banner component up more, and working to better mesh the parts together.
- Profile_About: Renders the user's biographical information and contact information, also subdivides the parent div.
    - **KNOWN BUG:** The CSS border elements fail to stretch all the way down to the end of the page and instead only stretch as far as writing/data appear. Some fiddling with CSS should resolve this, such as finding a way to stretch it to the end of its parent div. Hope to resolve before submission.
- Profile_Skills: Renders the user's skills and interest as a series of bootstrap buttons, subdivides the parent div.

- **KNOWN BUG:** Same as Profile_About above.
    - Profile_Teams: Renders the current and previous projects the user has worked on as a list of pills. It completes the subdivision of the parent div by Profile_About, Profile_Skills, and itself.
        - **KNOWN BUG:** Same as Profile_About and Profile_Skills above.

<u>Main Feed - Nicholas Achin</u>

The mount point appears on line 20 in */build/main-feed.html* as <...id="main-feed"> and the corresponding */app/components/main-feed.js* serves as the top level React component, calling on the subcomponents it imports and rendering appropriately.

Known Issues:
- The main-feed.html page does not handle shrinking of the browser horizontally well. We will add responsive design by adding CSS properties.
- The main feed section of the page does not fill the page with sufficient background coloring if there is a small amount of feed items on the page. This was working with static HTML before, but the transition to React muddied the waters. CSS fixes will remedy this.
- The filtering bar on the page needs to handle onChange events so that the page actually filters when boxes are checked/unchecked. Right now, the boxes won't uncheck within the React component. This required more work than originally planned so it will be handled in the next submission.
-

Components:

- The main feed utilizes the same Profile_Navbar and Profile_Sidebar components described above as these components are on every page of our application.
- MainFeed: This serves as the highest component calling on Profile_Navbar, Profile_Sidebar, MainFeedFilter, MainFeedJobItem, MainFeedProjectItem, and MainFeedNotificationItem to fill the main feed.
- MainFeedFilter: This contains the MainFeedFilter. Right now this component just renders the static HTML code. This component will be built out more and become more dynamic with more subcomponents.
- MainFeedJobItem: This is a feed item for the main feed that has a specific glyphicon that relates it to job posts on the main feed. These components also have a tag system with an underlying MainFeedTagList component.
- MainFeedNotificationItem: This is a feed item for the main feed that has a specific glyphicon that relates it to followed team posts on the main feed.
- MainFeedProjectItem: This is a feed item for the main feed that has a specific glyphicon that relates it to the user's teams on the main feed.

- MainFeedTagList: This is utilized for job posts to attach user's interests that the job opening possesses. The MainFeedTagList is composed of tag items that are dynamic for each job opening.
- MainFeedTagItem: These are simple dynamic button components that form a MainFeedTagList.

Inbox - Caroline Kim

The mount point appears on line 16 in */build/inbox.html* as <...id="inbox01">, in */build/inbox2.html* as <...id="inbox02">, and in */build/inbox3.html* as <...id="inbox03"> and the corresponding */app/components/inbox-inboxpage.js*, */app/components/inbox-inboxpage2.js*, and */app/components/inbox-inboxpage3.js* serves as the top level React component, calling on the subcomponents, it imports and rendering appropriately.

Known Issues:
- The window is made to fit into certain size of screen only. It either breaks or it's too small when the file is opened in a screen with different size. Using CSS to make the screen stretch to 100% of the window size should solve this problem.
- In the Inbox-panels, the glyphicons that indicate that this is a message box with more than one person should be next to the name, but it's on top of the name. Using CSS or changing up the div should solve this problem.
- In the InboxItems, the panels are too far away from each other. Using CSS should fix the margin to make it look cleaner.

Components:
- InboxHeader: Shows on top who you're currently messaging with
- Inbox: The panel on the left side of the page to show the names of people who messaged you
- InboxItems: Lists the props for Inbox panels
- InboxPage: The whole Inbox Page with InboxItems, MessageItems, sidebar, and top navbar
- InboxThread: Maps all the Inbox components to be stacked on top of another
- MessageEntry: The panel where the user writes down and sends the message
- MessageItems: Lists the props for messages and has MessageEntry on the bottom
- MessageReceive: Messages that was received from other users (left aligned)
- MessageSent: Messages that was sent by the user (right aligned)
- MessageThread: Maps all the MessageReceive and MessageSent components to be stacked on top of another

*There are 3 of each InboxPage, InboxItems, and MessageItems because they are all different for inbox.html, inbox2.html, and inbox3.html.

<u>Project Page - Thomas Cloutier</u>

The mount point appears on line 106 in */build/project-page.html* as <...id="project_page-zone"> and the corresponding */app/components/project-page.js* serves as the top level React component, calling on the subcomponents it imports and rendering appropriately.

Known Issues:
- The OSTF logo breaks the page right now so we have switched in a bugged home glyphicon right now. This will be fixed by the next submission.
- The apply and follow buttons currently do nothing, will be changed when implementing user interactivity
- Css will break at zoom greater than 150%

Components:
- ProjectPage : Component that displays all the information on a project to the user
- ProjectBanner : Displays the projects personalized banner with project title to the user
- ProjectDescription: Left-most div in the project page that holds all the projects description items
- ProjectMessages: Middle div in the project page that holds all the projects message items
- ProjectPositions: Right-most div in the project page that holds all the projects position related information
- MessageItem: Individual item message object displayed in the messages portion of the page
- SkillItem: Individual skill presented in the description portion of the page
- MemberItem: Individual project member information to be displayed in the description portion of the page
- PositionItem: Individual position information to be displayed in the position portion of the page

<u>Create Project Page - Ryan LeCours</u>
The mount point appears on line 120 in */build/createproj.html* as <...id="create-project"> and the corresponding */app/components/createproject.js* serves as the top level React component, calling on the subcomponents it imports and rendering appropriately.

Known Issues:

- The buttons are still set as dead links. These will be given functionality when we dynamically code our startup pages
- React has not been integrated with the side-bar and navigation bar as these are components on every page. Our team is planning to meet and discuss this to ensure continuity of each page.
- The site will break if you squeeze the page horizontally enough… although I do not believe this will be an issue graders will be concerned with because we did not cover responsive design in class. Regardless, I coded my entire page size-wise in percentages to make it as resilient as possible. Not really an issue as much as something just to be aware of on both the grader and team sides.
- May consider adding a position-section component because the position section will change as I add positions, however I will make that decision as we move further on with dynamic programming.

Components:
- <u>CreateProject</u>:
  Component that displays the main columns the user will be interacting with. Contains many input fields to take in the different information a project can contain.
- <u>CreateProjectPositionSection</u>:
  The component of the right-main column where the user can add positions, and add more slots for positions on a need-basis.
- <u>CreateProjectSinglePosition</u>:
  The components found inside the Position Section that actually handle the containing of a position needed for a team. Will be using this component when the "Add More…" button in the Position Section is given functionality.
- <u>CreateProjectTagItem</u>:
  A component saved as a single tag. This will be used to help users find a newly created project.
- <u>CreateProjectTagList</u>:
  A component containing all of the tags a project can have. Currently, the maximum size of this list will be five, but the number of tags in the list will change when we dynamically update these components
- <u>CreateProjectTagSection</u>:
  The component where a user can enter in tags, and upon hitting the "Add" button, the tags will get added to the Tag List. Tags will appear below the input box.

Job Board - Josh Tranfaglia

The mount point appears on line 99 in */build/ostf-job-board.html* as <...id="job-board-main-feed"> and the corresponding */app/components/job-board/job-board-main-feed.js* serves as the top level React component, calling on the subcomponents it imports and rendering appropriately.

Known Issues:
- The top navbar and the left side bar are currently not react components. This can be fixed with some refactoring, such as including another mount point higher up in the job board page to contain those two components.
- The list of tags in both the filter bar and each job board post are currently not react components. The tags will likely need some form of refactoring once data is ready to be mocked to the page. For now, they take the form of hyperlinks, one after another in the html. Once the data is ready, the tags can be properly replaced with react components.
- The recommendation level medal icon is currently not a react component. This should be changed by installing the classnames package with npm and adding the proper color of the recommendation as a react prop to the recommendation, but we experienced some issues when implementing this and had to drop it until after the deadline.

Components:
- FilterBar:
    - Component the user interacts with to filter specific tags on the job board.
- JobBoardMainFeed:
    - Component used to render the entirety of the main feed of the job board, including the filter bar.
- JobBoardPost:
    - Component used to display each individual job posting, including those searched and those recommended, to the user.