Founders:
Jeremy Lee
SeihakRithy Muth
Seyedeh Arta Razavi
Brendan Shaughnessy
Colin Stern
David Zagreda

Rithy's Components:

Matched:
This is the main component of the matched page that contains other subcomponents. It is also the component that get mounted onto the html page *matched.html*.

Matched_User:
This is the user component of the matched page. It renders the two users trading and contains a subcomponent of the items being trade.

Matched_User_Item:
This is the item component of the matched page. It renders the the items being trade inside the user component.

Arta's Components:

Home:
This is the main component of the home page that contains other subcomponents. It is also the component that get mounted onto the html page home.html.
Bugs: No obvious bugs.

Home_Feed:
This is the feed component of the home page. It renders the list of available items for trade and contains subcomponents of the items available for trade.
Bugs: No obvious bugs.

Home_Side_Bar:
This is the sidebar component of the home page. It renders the user name and user image and contains subcomponents of the items the user has available for trade.
Bugs: No obvious bugs.

Home_Feed_Item:
This is a subcomponent of the feed item and side bar of home page. feed item renders the postings made users which includes book image, name, author, edition, isbn, and publisher.
Bugs: No obvious bugs.

<u>Navbar:</u>
This is the navbar component for all the pages. It renders a navbar on top of the page with a search field and the company logo.
Bugs: No obvious bugs.

David's Components:

<u>Message:</u>
This is the main component of the message page that contains other subcomponents. This is the component that gets mounted onto the html page message.html.

Bugs: Sidebar users are statically defined in message.js. To fix this, I should use the map function to render each child into a list.

<u>Message_Sidebar_User:</u>
This is the sidebar user component of the message page. This renders users with whom you are speaking, and includes fields such as an avatar, name, timestamp of last message, and the number of unread messages.

<u>Message_Main_User:</u>
This is the user component for the main feed of the message page. This component renders the users who are currently talking, their respective avatars, a timestamp of each message, and the actual message that they are sending.

Colin's Components:

<u>Search:</u>
This is the search component, the main component of the search page. It gets mounted onto search.html. It displays a search feed with padding between search feed items.
Bugs: Necessary to include js/app.js in search.html in order for it to work. Navbar says "Search textbooks," but there are options in my left sidebar to allow for searching other types of items.

<u>Search_Feed:</u>
This is the search feed component. It represents a search feed with a list of search items.
Bugs: No obvious bugs.

<u>Search_Feed_Item:</u>
This is the search feed item component. It represents a search feed item with a picture, title. author, location.

Bugs: No obvious bugs.

Brendan's Components:

CreateListing:
This component is the container for the different listing items. It contains the listing information options class and is mounted into the CreateListing html page.

ListingInfo:
This component contains the form options for what is allowed to trade. For example categories of items and class subjects. This is a child of two container classes which are for the item you are listing and the item you are looking for.

YourItem:
This class holds an upload image and listing title for your item. It also holds a child for listing info that is shared between this class and the item you are looking for.

SearchItem:
This class only holds a title and a child which will be listing info. All for the item you are looking for.

Jeremy's Components:

Userprofile:
This is the static react page with all of the subcomponents for the userprofile page.

Userprofile_Panel:
This is the subcomponent of userprofile that conglomerates the main panel of the userprofile page. The panel contains user listings of textbooks the user needs and the textbooks the user has. It represents a search feed item with a picture, title. author, location.
Bugs: There are no usages of props for the attributes of the textbooks such as Author, ISBN number, Textbook name, etc.

Userprofile_Sidebar:
This is the subcomponent of the profile sidebar section of the userprofile. It contains the user profile's with a picture, title. and links to edit profile, look at the user's listings, check the user's matches, and an additional help link.

Bugs: If the sidebar was given data, then the data itself will not render. It will render the text that was initially put in during the static HTML/CSS startup assignment. It will be fixed in time for the next startup.

- As of now, we have decided to not cut any product features from our application.