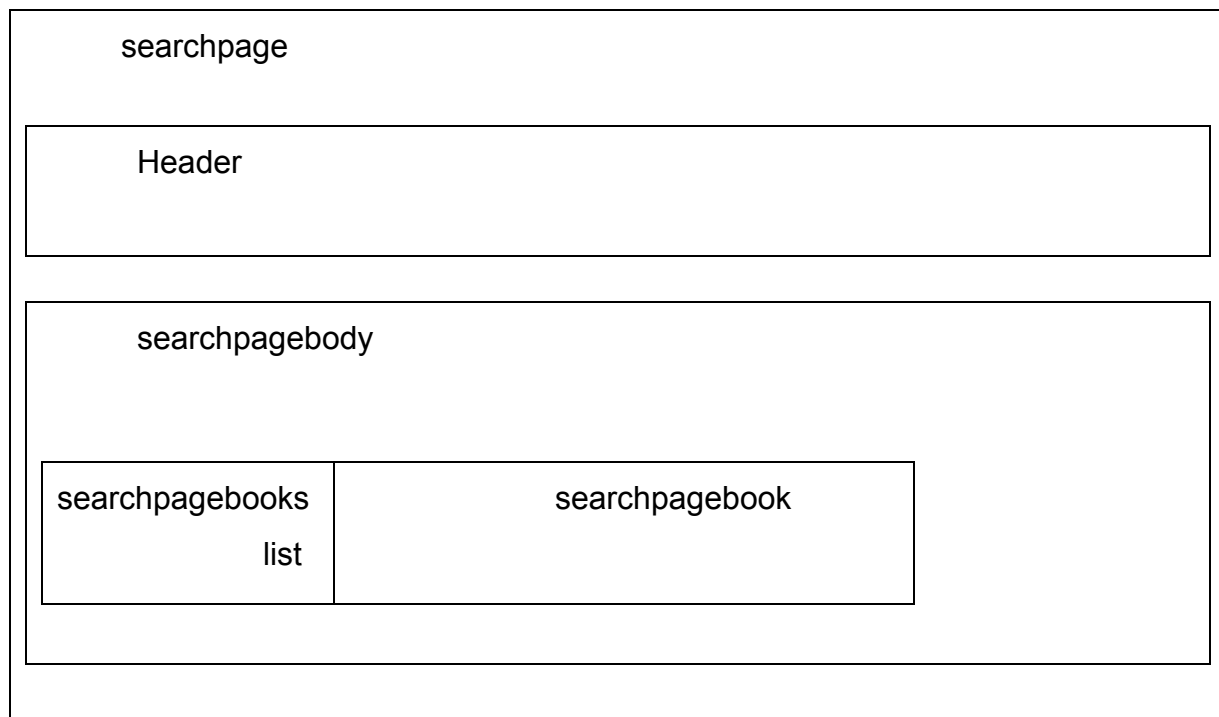


Dynamic UI Mock-Up Report

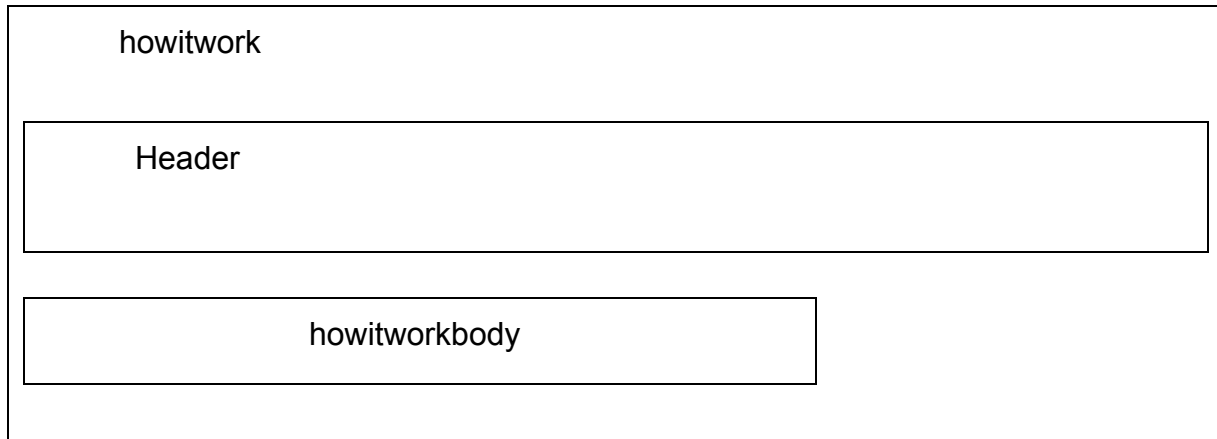
Team members: Zhuodong Huang(28261551), Ye Jiang,
Zhongce Ji(28551884), Kai Mastuda (26414899)

- header.js: Component includes the website's nav bar and banner. The nav bar includes links to other parts of the site and a search bar, which will navigate to the search page. All pages will include the header component.
- searchpage.js: Main react component for the search page. It contains two components, which are header.js and searchpagebody.js. The main feed feature will display inside the body component. Also, the body component break down into searchpagebook.js and searchpagelist.js component. Each searchpagebook component will display different books from the database. The searchpagebookslist component, which displays the browsing history.

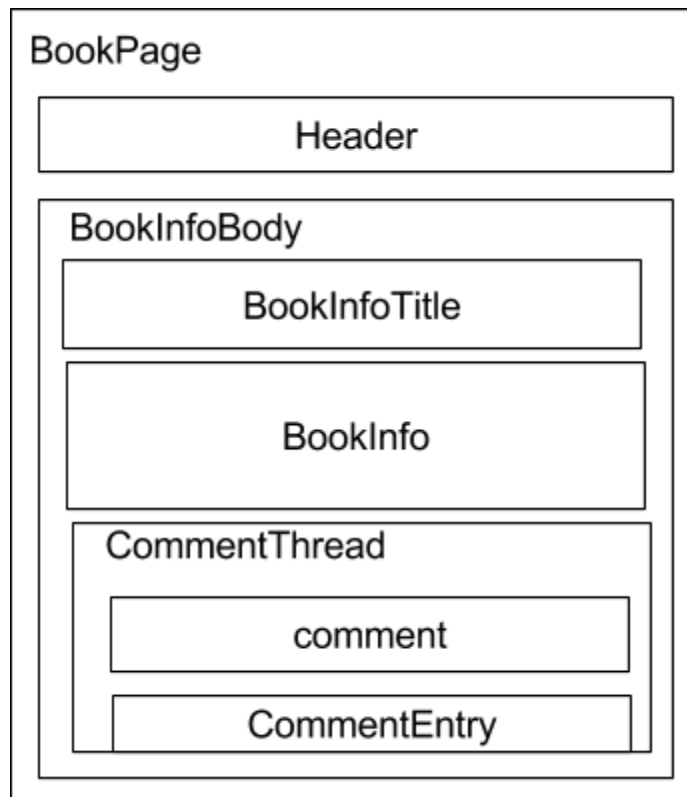


- howitwork.js: Main react component for howitwork page. It contains header and howitworkbody components. Since howitwork page is only for help and contact, we don't need to get data from database. Therefore, we don't need to nest any

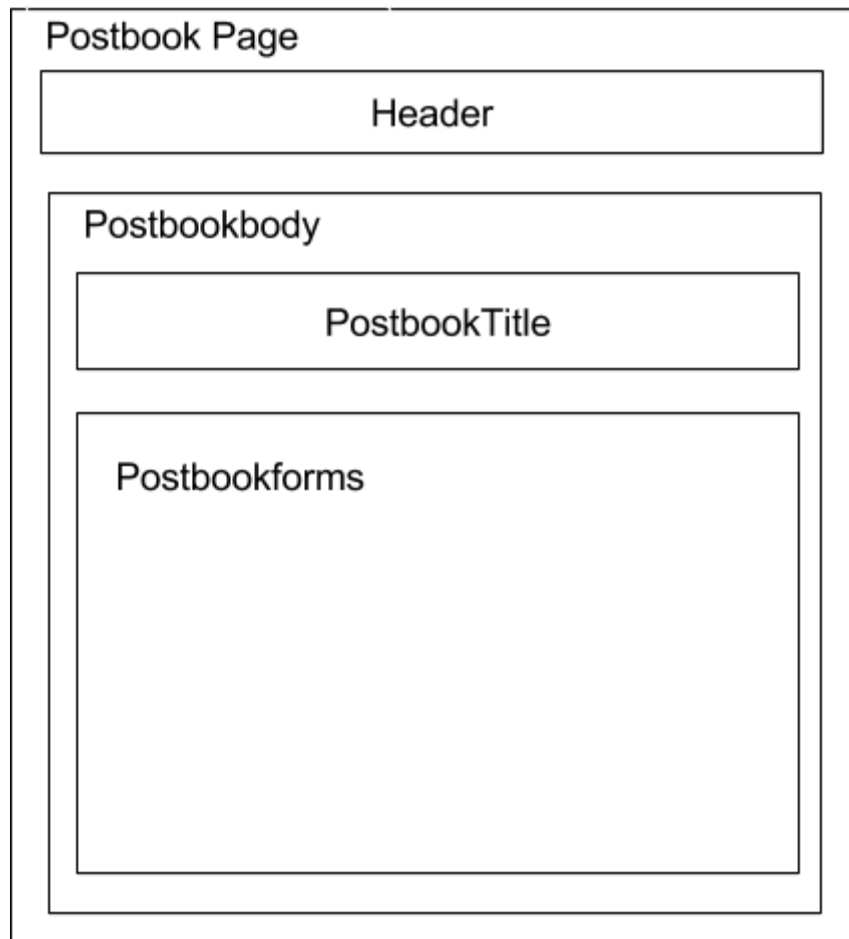
components. In the howitwork page, we also build a link to an email that the user can use to contact us.



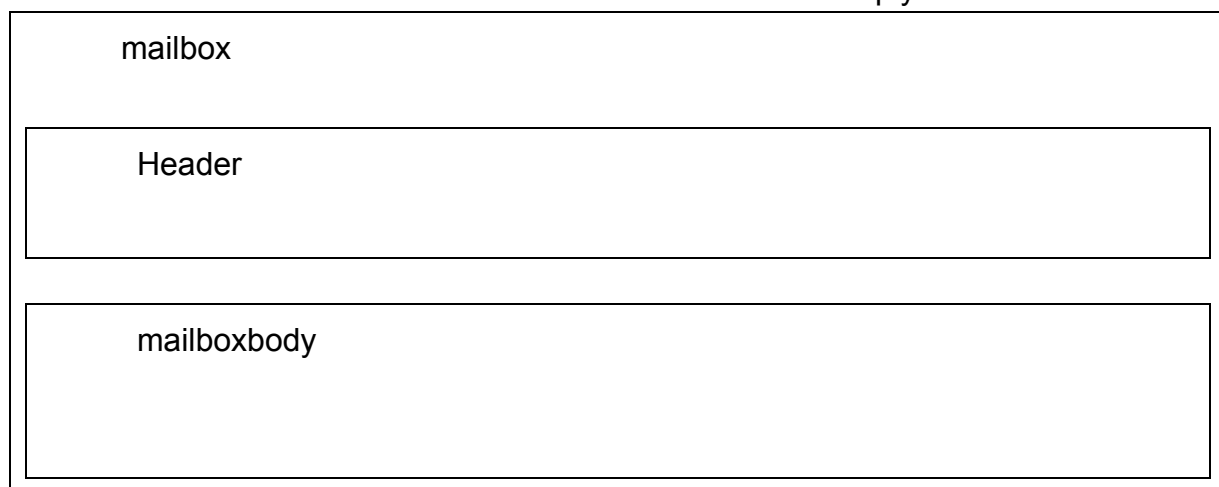
- bookinfo.js: The bodyinfobody has the bookinfotitle, bookinfo, commentthread components. The commentthread component has the comment and commententry components. The comment entry allows for the user to type a comment for a book and submit to the database. The bookinfo component holds the information about the book. The comment component will show all the comments the book has. The back to search link in the booinfobody will take you back to the search page. The user name will take you to the user profile page. The buy book button will take you to the page where the page says the book is sold out and there is another button that can take you back to search page

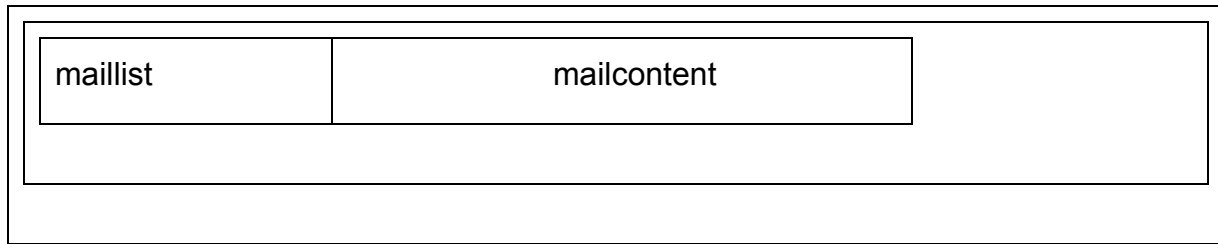


- postbookpage.js: The user can post a book using this page. The user can type all the information about the book and click submit; the book will then be added into the database. The page will then go to the page which says the book posted successfully (there is a button that will take you back to the search page). And the search page list will then show the book the user has posted. The pic field is fixed with the router of the first book picture for now and we will change it at a later date when we are more knowledgeable of the given topic. The cancel button will then take you back to the front page.

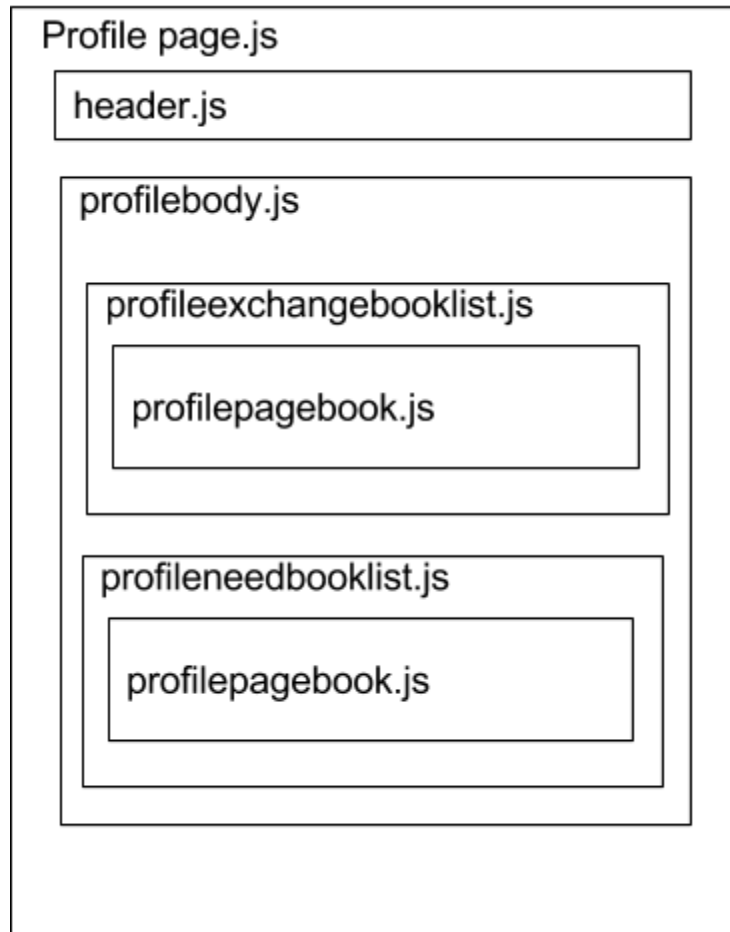


- mailbox.js: Contains the mailboxbody component which holds the content of the user's mailbox. The mailboxbody contains two components: the maillist, which displays the list of active convos the user is a part of, and mailcontent, which displays the content of one of the active convos the user has selected. The mailcontent also contains a form field for the user to reply to the convo with.





- profile.js: The main component for profile page, which include header and profile body. In the profile page will pass the user id to the profilebody so that the profilebody can dereference the user data to show all the informations.
- profilebody.js: The main feed for the user information will display in this component.. The page will show the name of the user, how many books the user need, how many book the user want to exchange, the location where the user are and also how many message they got from others. It breaks down into profileexchangebooklist and profileneedbooklist components. They will get data from database, shows what books user currently need and exchange. Each list component will contain profilepagebook components, which use to display the book information.



Features Cuts:

1. remove homepage, use frontpage as home page.
2. remove hot list feature(may add it back in the future)
3. UBooks will focus on exchange for now. Remove buy or sell.(may add it back in the future)

Known Bugs:

- search result is just mock up, which display all the books from database. not sure how to transmit between two separate components.
- Browsing history is just mock up, which is just random display a book from database.
- Mailbox is just mock up to talk to someone. It is not support to talk to many different people for now.

Contributions:

- Everyone contributed to server.js and database.js
- Zhuodong Huang:

- header.js
 - searchpage.js
 - howitwork.js
- Zhongce Ji
 - bookinfopage.js
 - postbookpage.js
- Kai Matsuda
 - mailbox.js
 - report
- Carter Jiang
 - frontpage.js
 - personalprofilepage.js