Test case for mailstone3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test file | Covered | Test time | Result | Test date | Expected result | How did we solve that |
| App.test.js | Function testing of each button and sidebar button works on App.js | 0.4 | passed | 10/21 | All the button could be click and lead to each page and index | nah |
| Book.test.js | Test the front end would be able to touch the back-end with the database of books | 0.3 | failed | 10/15 | Able to connect the database | Fix the function of api connection and created an jason file with serial data (fixed) |
| Header.test.js | The headers and landings are involved in each page | 0.2 | failed | 10/16 | All pages should have headers and landing | This is an mistake testing, because not every js file should have index.(fixed) |
| Sidersearch.test.js | A specific js file contains the function of search the book | 0.3 | Failed | 10/21 | The sidebar should be able to search the information from the database, and output the expect answer | We change the direction from data.jason to the database |
| Addperson.test.js | Able to add person to the database(front-end and back-end connection ) | 0.2 | passed | 10/15 | Able to created account for customer and able to login in | Nah |
| Componeent.test.js | Get the text of the app | 0.2 | faild | 10/15 | App should show up what we’ve write on that page | The web is working but the test is failed, so we are fixing the code of test.js |
| Soldbook.test.js | Are able to see all the books that we have on database | 0.65 | failed | 10/15 | Should able to see the books inside describe(“filter function ”) | We can see through the web test. But the jest test was failed, the function is actually worked. |
| Setup tets.js | Import the testing library with jest-dom | 0.1 | Passed | 10/22 | Import the jest-dom for the react test | Nah |
|  |  |  |  |  |  |  |