Unit test report:

```
 PASS   src/setupTest.test.js
  Test Auth
    ✓ should log in a previously registered user (144 ms)
    ✓ should not log in an invalid user (26 ms)
    ✓ should log out a user (9 ms)
  Test Article
    ✓ should fetch articles for current logged in user (194 ms)
    ✓ should filter displayed articles by the search keyword (155 ms)
    ✓ should add articles when adding a follower (182 ms)
    ✓ should remove articles when removing a follower (97 ms)
  Test Profile
    ✓ should fetch the user's profile username (135 ms)


Test Suites: 1 passed, 1 total
Tests:       8 passed, 8 total
Snapshots:   0 total
Time:        4.001 s
```

Test coverage report:

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	56.29	39.75	54.32	53.7	
src	0	0	0	0	
App.js	0	100	0	0	8
index.js	0	100	100	0	7-17
reportWebVitals.js	0	0	0	0	1-8
src/components	52.25	61.76	37.14	51.81	
LoginForm.js	91.42	85.71	75	91.17	12-14,25
MainHeader.js	87.5	100	66.66	87.5	7
PostGrid.js	42.85	0	20	42.85	8-21,24-25,34-40
Posts.js	62.5	83.33	40	62.5	30-32
ProfileHeader.js	75	100	50	75	5
RegistrationForm.js	0	0	0	0	3-53
UserInfoGrid.js	45.45	50	33.33	45.45	9-13,28
src/pages	63.19	26.66	72.09	59.45	
Landing.js	0	100	0	0	7-10
Main.js	95.12	73.33	88.23	94.02	29-30,109,138
Profile.js	32.05	3.33	12.5	32.05	35-105,132-162

```
Test Suites: 1 passed, 1 total
Tests:       8 passed, 8 total
Snapshots:   0 total
Time:        4.65 s
```

Note that the files that contain application logics are:
- src/components/loginForm.js   - 91.17%
- src/components/MainHeader.js - 87.5%
- src/pages/Main.js   - 94.02 %
- src/pages/Profile.js   - 32.05%

The rest of the files in the above screenshot mainly contain html code. I also want to point out that since my Profile.js and MainHeader.js (especially Profile.js) also contain more html than other logic files, they have a lower coverage rate.

Overall, I think the combined coverage should reach 90% not considering html code.