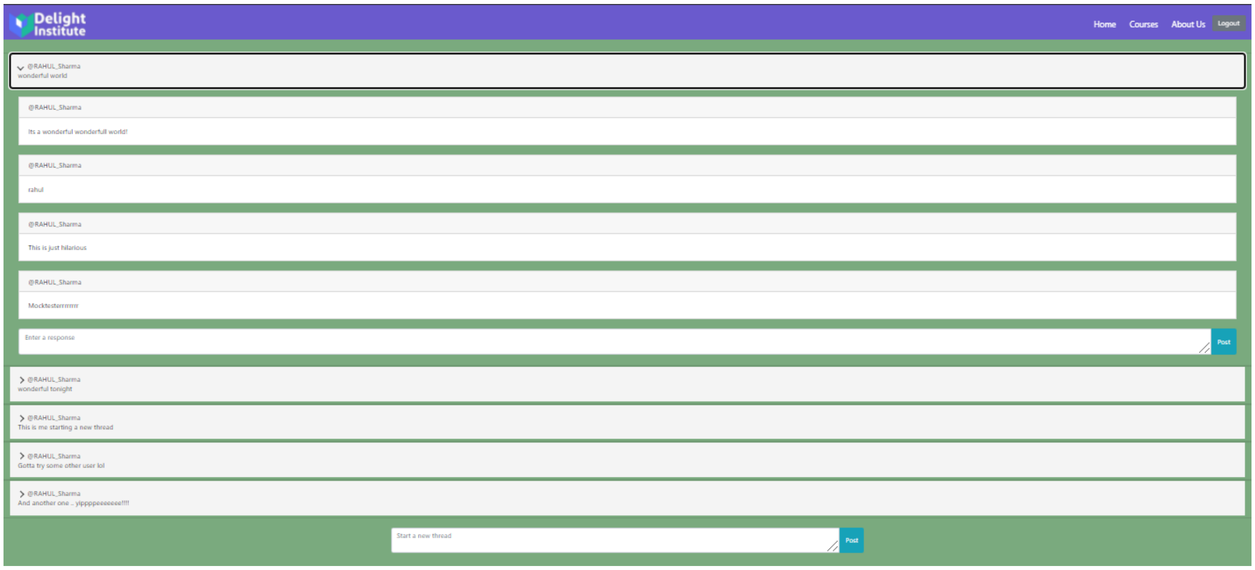
Delight Institute

Introduction

This project aims at designing a website for an institution named delight institute for taking online examinations. A system for conducting mock exams for people to prepare for big exams or school exams is the most mandatory feature of this website. Other than that, it has a dashboard for teachers and professors, which opens depending on the login id. The teacher can create courses and add exam questions to existing question. The student may buy a course from the courses page. An additional discussion forum will also be provided for the students and teachers to discuss their views and solutions to different problems.

Design and Implementation

- Before the actual code was written, a detailed flow and wireframe diagrams were made to specifically understand where to put different components on the screen.
- Redux as a state management library has NOT been used since the project was not huge and Redux added more than a few extra lines which increased the code size. Rather MobX, a state management library was used as using it is easier. To persist form states, MobX-persistance library was used.
- Font awesome and Reactstrap was heavily used in overall construction of different components as it took out much of the CSS complexity out of question. More functional components were used rather than class components as react hooks like useEffect and useState make life easier.
- ScreenShots:



@RAHUL_Sharma
wonderful world

@RAHUL_Sharma
Its a wonderful wonderfull world!

@RAHUL_Sharma
rahul

@RAHUL_Sharma
This is just hilarious

@RAHUL_Sharma
Mocktesterrrrrrr

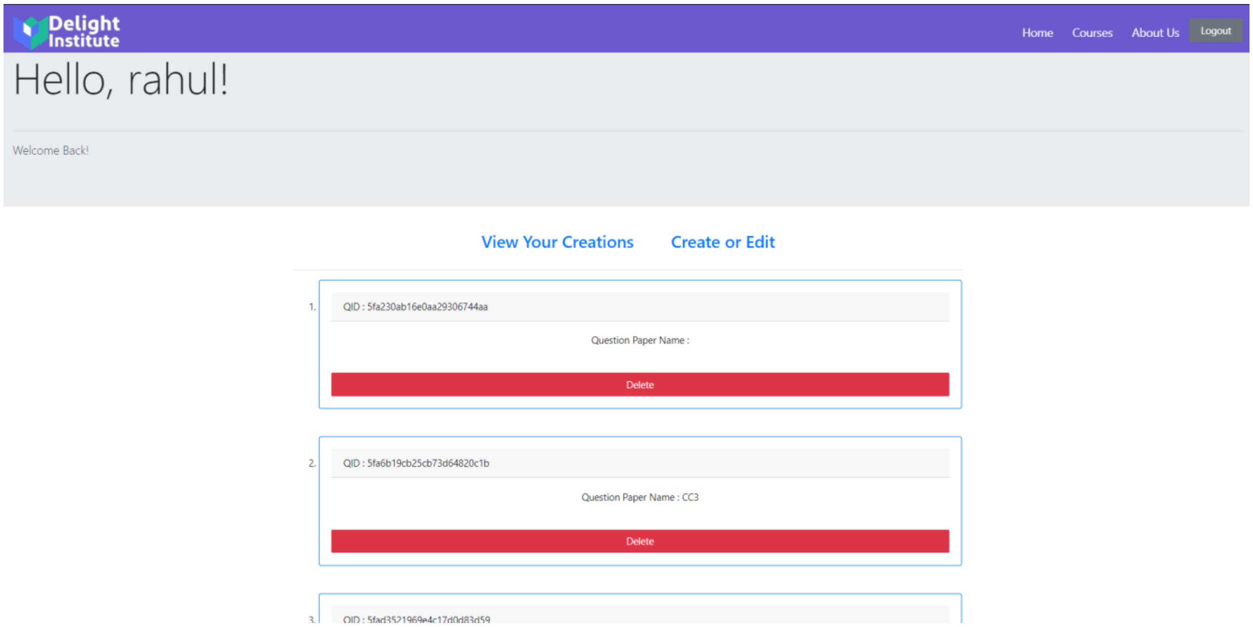
@RAHUL_Sharma
wonderful tonight

@RAHUL_Sharma
This is me starting a new thread

@RAHUL_Sharma
Gotta try some other user lol

@RAHUL_Sharma
And another one .. yipppppeeeeeee!!!!

The comment section design



Hello, rahul!

Welcome Back!

View Your Creations   Create or Edit

1. QID : 5fa230ab16e0aa29306744aa

Question Paper Name :

Delete

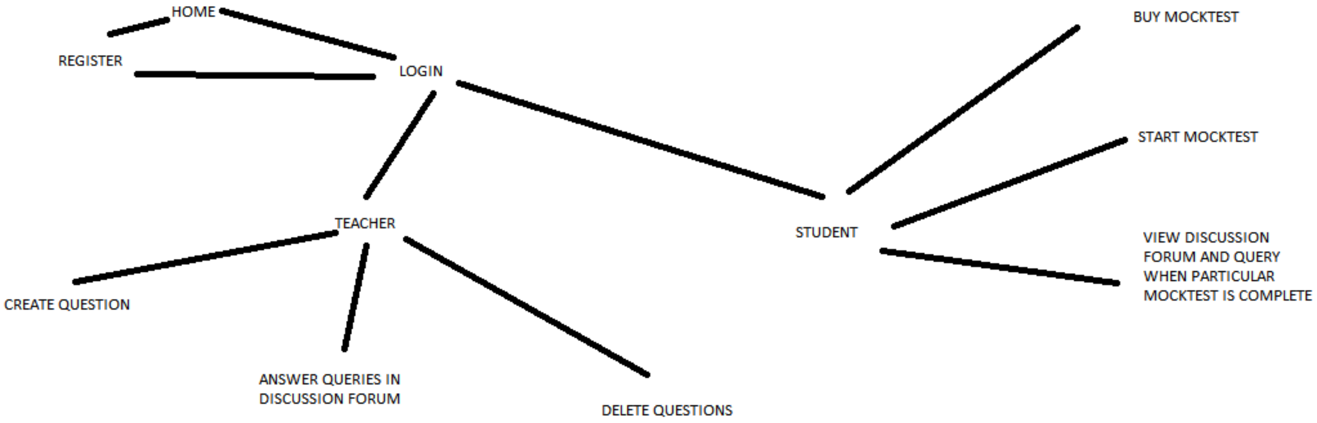
2. QID : 5fa6b19cb25cb73d64820c1b

Question Paper Name : CC3

Delete

3. QID : 5fad3521969e4c17d0d83d59

The page where a teacher views the question templates he created



And so on and so forth …

Conclusion

- Redux does add to the code, but it cannot be fully replaced as the other state management libraries do not include or have proper documentation or have community for the usage of the software and the errors that happen.
- Transitions and animations have not been added to the project. To add to the user experience, this is utterly necessary. The spinners in case of async actions need to placed, to show the on load event, but its tough to achieve such a task without redux, due to lack of proper documentation on proper use of mobx-async.
- The react-router libraries on the newer versions, have a problem that the link in the address bar changes but the component does not change view, or the next component is not mounted or the existing component is not updated till the reload button is clicked. But on forceRefresh on the Browser Router the entire state of the store and every other state is reset. This is a a bug in the library itself. The last stable version seems to 4.2.2. This can be overcome by using the localstorage of the browser itself and making each page an app in itself. Though the workaround adds a bit of extra code and check functions, it does work.

- If I had to undo the entire project, I would probably use react-router@4.2.2 , use redux as a state management library, and include more animations into the project.

References

www.coursera.org

www.udemy.com