

Exercise (Instructions): React-Redux-Form Revised

Objectives and Outcomes

In this exercise we will explore the interaction between react-redux-form and the Redux store. We will see how to map the form into the store so that the state of the form will be persisted in the store. At the end of this exercise you will be able to:

- Use react-redux-form to interact with Redux store and store the state of the form in the store.

Updating the Feedback Form

- Add a new file named `forms.js` in the `redux` folder and add the following to it:

```
1 export const InitialFeedback = {
2   firstname: '',
3   lastname: '',
4   telnum: '',
5   email: '',
6   agree: false,
7   contactType: 'Tel.',
8   message: ''
9 };
```

- Then, open `configureStore.js` and update it to add the form to the reducers:

```
1 ...
2
3 import { createForms } from 'react-redux-form';
4 ...
5 ...
6 ...
7 import { InitialFeedback } from './forms';
8 ...
9 ...
10 ...
11 combineReducers({
12   dishes: Dishes,
13   comments: Comments,
14   promotions: Promotions,
15   leaders: Leaders,
16   ...createForms({
17     feedback: InitialFeedback
18   })
19 },)
20 ...
21 ...
```

- Next, open `MainComponent.js` and update it as follows:

```
1 ...
2
3 import { actions } from 'react-redux-form';
4 ...
5 ...
6 ...
7 resetFeedbackForm: () => { dispatch(actions.reset('feedback'))}
8 ...
9 ...
10 ...
11 <Route exact path="/contactus" component={() => <Contact
12   resetFeedbackForm={this.props.resetFeedbackForm} /> />
13 ...
```

- Open `CommentComponent.js` and update it as follows:

```
1 ...
2
3 import { Control, Form, Errors, actions } from 'react-redux-form';
4 ...
5 ...
6 ...
7 handleSubmit(values) {
8   console.log('Current State is: ' + JSON.stringify(values));
9   alert('Current State is: ' + JSON.stringify(values));
10   this.props.resetFeedbackForm();
11   // event.preventDefault();
12 }
13 ...
14 ...
15 ...
16 <Form model="feedback" onSubmit={(values) => this
17   .handleSubmit(values)}>
18   ...
19   ...
20   ...
21 </Form>
22 ...
23 ...
```

- Save all the changes and do a Git commit with the message "React Redux Forms Revised".

Conclusions

In this exercise we have seen how to use react-redux-form together with Redux to persist form state.

Mark as completed

