

MỤC TIÊU:

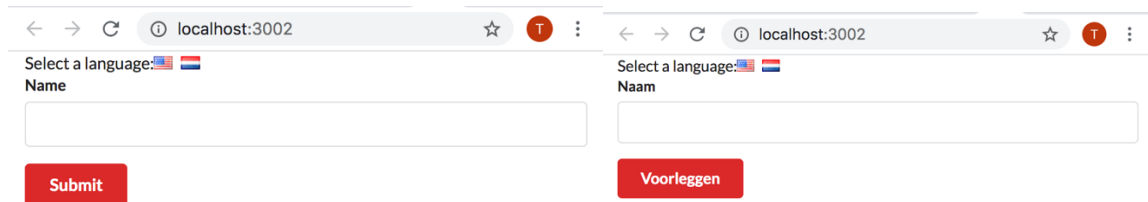
Kết thúc bài thực hành này bạn có khả năng

- ✓ Context API.
- ✓ React với firebase

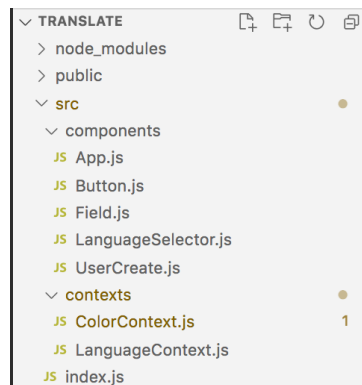
NỘI DUNG

Bài 1: Context API

- Tạo project translate có cấu trúc, mô tả và hướng dẫn sau
- Mô tả



- Cấu trúc



- Hướng dẫn

```
src > components > JS App.js > ...
1 > import React from 'react'; ...
6
7 class App extends React.Component {
8   render() {
9     return (
10       <div className="ui container">
11         <LanguageStore>
12           <LanguageSelector />
13
14           <ColorContext.Provider value="red">
15             <UserCreate />
16           </ColorContext.Provider>
17         </LanguageStore>
18       </div>
19     );
20   }
21 }
22
23 export default App;
```

```
src > components > JS Button.js > ...
1 > import React from 'react'; ...
4
5 class Button extends React.Component {
6   renderSubmit(language) {
7     return language === 'english' ? 'Submit' : 'Voorleggen';
8   }
9
10  renderButton(color) {
11    return (
12      <button className={`ui button ${color}`}>
13        <LanguageContext.Consumer>
14          | ({ language }) => this.renderSubmit(language)
15        </LanguageContext.Consumer>
16      </button>
17    );
18  }
19
20  render() {
21    return (
22      <ColorContext.Consumer>
23        | {color => this.renderButton(color)}
24      </ColorContext.Consumer>
25    );
26  }
27 }
28
29 export default Button;
```

```
src > components > JS Field.js > ...
1 > import React from 'react'; ...
3
4 class Field extends React.Component {
5   static contextType = LanguageContext;
6
7   render() {
8     const text = this.context.language === 'english' ? 'Name' : 'Naam';
9
10    return (
11      <div className="ui field">
12        <label>{text}</label>
13        <input />
14      </div>
15    );
16  }
17 }
18
19 export default Field;
```

```
src > components > JS LanguageSelector.js > ...
```

```
1 > import React from 'react'; ...
3
4 class LanguageSelector extends React.Component {
5   static contextType = LanguageContext;
6
7   render() {
8     return (
9       <div>
10        Select a language:
11        <i
12          className="flag us"
13          onClick={() => this.context.onLanguageChange('english')}
14        />
15        <i
16          className="flag nl"
17          onClick={() => this.context.onLanguageChange('dutch')}
18        />
19      </div>
20    );
21  }
22 }
23
24 export default LanguageSelector;
25
```

```
src > components > JS UserCreate.js > ...
```

```
1 import React from 'react';
2 import Field from './Field';
3 import Button from './Button';
4
5 const UserCreate = () => {
6   return (
7     <div className="ui form">
8       <Field />
9       <Button />
10    </div>
11  );
12 };
13
14 export default UserCreate;
15
```

```
src > contexts > JS ColorContext.js > ...
```

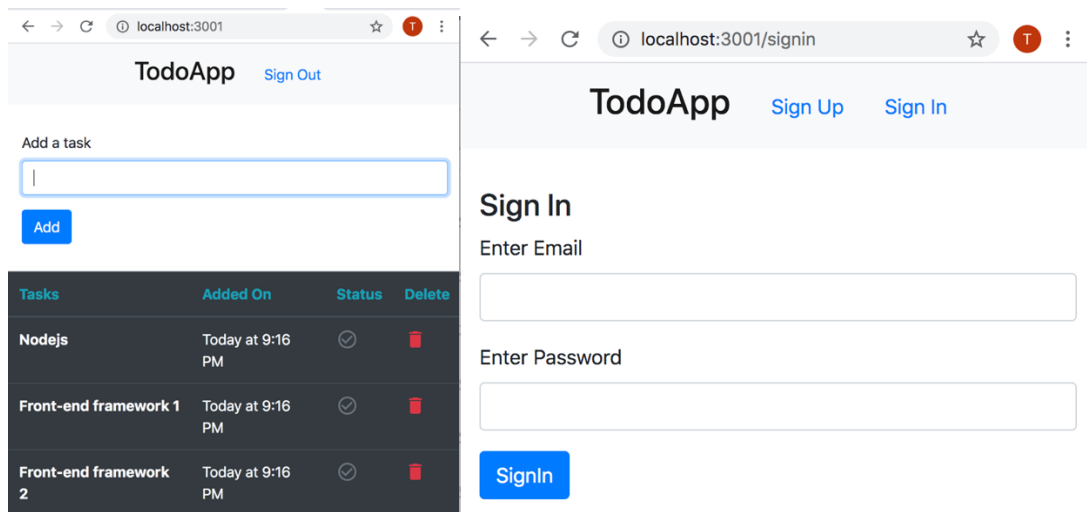
```
1 import React from 'react';
2
3 export default React.createContext();
4
```

```
src > contexts > JS LanguageContext.js > ...
```

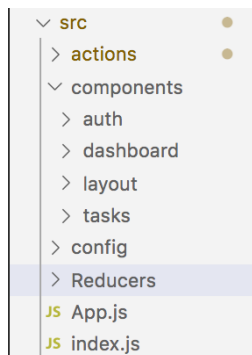
```
1 import React from 'react';
2
3 const Context = React.createContext('english');
4
5 export class LanguageStore extends React.Component {
6   state = { language: 'english' };
7
8   onLanguageChange = language => {
9     this.setState({ language });
10  };
11
12   render() {
13     return (
14       <Context.Provider
15         value={{ ...this.state, onLanguageChange: this.onLanguageChange }}
16       >
17         {this.props.children}
18       </Context.Provider>
19     );
20  }
21 }
22
23 export default Context;
24
```

Bài 2: react redux firebase

- Tạo project ReactFire có mô tả, cấu trúc và hướng dẫn sau
- Mô tả



- Cấu trúc



- Hướng dẫn

```
src > JS index.js > ...
1 > import React from "react"; --
14
15 const store = createStore(
16   rootReducer,
17   applyMiddleware(thunk.withExtraArgument({ getFirebase }))
18 );
19
20 const rrfProps = {
21   firebase,
22   config: {},
23   dispatch: store.dispatch,
24   createFirestoreInstance,
25 };
26
27
28
29 ReactDOM.render(
30   <Provider store={store}>
31     <ReactReduxFirebaseProvider {...rrfProps}>
32       <App />
33     </ReactReduxFirebaseProvider>
34   </Provider>,
35   document.getElementById("root")
36 );
37
38
39
```

```
src > JS App.js > ...
1 > import React from "react"; ...
9
10 function App() {
11   return (
12     <>
13       <BrowserRouter>
14       <ToastContainer />
15       <NavBar />
16       <Switch>
17         <Route path="/signin" component={SignIn} />
18         <Route path="/signup" component={SignUp} />
19         <Route path="/" component={Dashboard} />
20       </Switch>
21     </BrowserRouter>
22   </>
23 );
24 }
25
26 export default App;
```

- Hướng dẫn: components

```
src > components > dashboard > Dashboard.jsx > ...
1 import React from "react";
2 import AddTask from "../tasks/AddTask";
3 import Tasks from "../tasks/Tasks";
4 import { connect } from "react-redux";
5 import { Redirect } from "react-router-dom"
6
7 const Dashboard = ({ uid }) => {
8   if (!uid) return <Redirect to="/signin" />;
9   return (
10     <>
11       <AddTask />
12       <Tasks />
13     </>
14   );
15 };
16
17 const mapStateToProps = (state) => {
18   const uid = state.firebase.auth.uid;
19   return {
20     uid: uid,
21   };
22 };
23
24 export default connect(mapStateToProps)(Dashboard);
```

src > components > layout > NavBar.jsx > default

```

1  import React from "react";
2  import { Link } from "react-router-dom";
3  import NavItems from "../NavItems";
4  import { connect } from "react-redux";
5
6  const NavBar = () => {
7    return (
8      <>
9        <nav className="navbar navbar-light bg-light d-flex justify-content-center">
10         <Link to="/" className="navbar-brand">
11           <h3>TodoApp</h3>
12         </Link>
13         <NavItems />
14       </nav>
15     </>
16   );
17 };
18
19 const mapStateToProps = (state) => {
20   //console.log(state);
21   return {};
22 };
23
24 export default connect(mapStateToProps)(NavBar);
25

```

src > components > layout > NavItems.jsx > mapStateToProps

```

1  import React from "react";
2  import { Link } from "react-router-dom";
3  import { signOut } from "../../actions/authActions";
4  import { connect } from "react-redux";
5
6  const NavItems = ({ signOut, uid }) => {
7    if (uid) {
8      return (
9        <Link to="/signin" className="nav-link" onClick={signOut}>
10         Sign Out
11       </Link>
12     );
13   } else {
14     return (
15       <>
16         <Link to="/signup" className="nav-link">
17           Sign Up
18         </Link>
19         <Link to="/signin" className="nav-link">
20           Sign In
21         </Link>
22       </>
23     );
24   }
25 };
26
27 const mapStateToProps = (state) => {
28   const uid = state.firebase.auth.uid;
29   return {
30     uid: uid,
31   };
32 };
33
34 const mapDispatchToProps = (dispatch) => {
35   return {
36     signOut: () => dispatch(signOut()),
37   };
38 };
39
40 export default connect(mapStateToProps, mapDispatchToProps)(NavItems);
41

```

```
src > components > auth > SignIn.jsx > ...
```

```

1  import React, { Component } from "react";
2  import { signIn } from "../../actions/authActions";
3  import { connect } from "react-redux";
4  import { Redirect } from "react-router-dom";
5
6  class SignIn extends Component {
7    state = {
8      email: "",
9      password: "",
10   };
11
12   handleChange = (e) => {
13     this.setState({
14       [e.target.id]: e.target.value,
15     });
16   };
17
18   handleSubmit = (e) => {
19     //console.log("SUBMIT");
20     //console.log(this.state);
21     e.preventDefault();
22     this.props.signIn(this.state);
23   };

```

```
src > components > auth > SignIn.jsx > ...
```

```

24
25   render() {
26     const { uid } = this.props;
27     if(uid) return <Redirect to="/" />
28     return (
29       <>
30         <form
31           className="container"
32           autoComplete="off"
33           style={{ marginTop: "30px" }}
34           onSubmit={this.handleSubmit}
35         >
36           <legend>
37             { " " }
38             <h4>Sign In</h4>
39           </legend>
40           <div className="form-group">
41             <label htmlFor="email">Enter Email</label>
42             <input
43               type="email"
44               className="form-control"
45               id="email"
46               onChange={this.handleChange}
47             />
48           </div>
49           <div className="form-group">
50             <label htmlFor="password">Enter Password</label>
51             <input
52               type="password"
53               className="form-control"
54               id="password"
55               onChange={this.handleChange}
56             />
57           </div>
58           <button type="submit" className="btn btn-primary">
59             SignIn
60           </button>
61         </form>
62       </>
63     );
64   }
65 }
66

```

```
src > components > auth > SignIn.jsx > ...
60
67 const mapStateToProps = (state) => {
68   console.log(state);
69   const uid = state.firebase.auth.uid;
70   console.log(uid);
71   return {
72     uid: uid,
73   };
74 };
75
76 const mapDispatchToProps = (dispatch) => {
77   return {
78     signIn: (creds) => dispatch(signIn(creds)),
79   };
80 };
81
82 export default connect(mapStateToProps, mapDispatchToProps)(SignIn);
83
```

```
src > components > tasks > AddTask.jsx > ...
1 > import React, { Component } from "react";-
4
5 class AddTask extends Component {
6   state = {
7     task: "",
8     checked: "false",
9   };
10
11   handleChange = (e) => {
12     this.setState({
13       [e.target.id]: e.target.value,
14     });
15   };
16
17   handleSubmit = (e) => {
18     e.preventDefault();
19     this.props.addTask(this.state);
20     document.getElementById("addTaskForm").reset();
21     console.log(this.state);
22   };
23
24   render() {
25     return (
26       <form
27         id="addTaskForm"
28         className="container"
29         autoComplete="off"
30         style={{ marginTop: "30px" }}
31         onSubmit={this.handleSubmit}
32       >
33         <legend> </legend>
34         <div className="form-group">
35           <label htmlFor="task">Add a task</label>
36           <input
37             type="text"
38             className="form-control"
39             id="task"
40             onChange={this.handleChange}
41           />
42         </div>
43         <button type="submit" className="btn btn-primary">
44           Add
45         </button>
46       </form>
47     );
48   };
49 }
50
51 }
52
53 const mapDispatchToProps = (dispatch) => {
54   return {
55     addTask: (task) => dispatch(addTask(task)),
56   };
57 };
58
59 export default connect(null, mapDispatchToProps)(AddTask);
60
```



```
src > components > tasks > Task.jsx > ...
1 > import React from "react"; ...
7
8 const Task = ({ task, removeTask, toggleChecked }) => {
9   const handleRemove = (task) => {
10     removeTask(task);
11   };
12
13   const handleCheck = (task) => {
14     toggleChecked(task);
15   };
16
17   return (
18     <>
19       <tr>
20         <th>{task.task}</th>
21         <td>{moment(task.date.toDate()).calendar()}</td>
22         <td>
23           <Check onClick={() => handleCheck(task)} checked={task.checked} />
24         </td>
25         <td>
26           <span
27             className="material-icons text-danger"
28             style={{ cursor: "pointer" }}
29             onClick={() => handleRemove(task)}
30           >
31             delete
32           </span>
33         </td>
34       </tr>
35     </>
36   );
37 };
38
39 const mapDispatchToProps = (dispatch) => {
40   return {
41     removeTask: (task) => dispatch(removeTask(task)),
42     //toggleChecked: (task) => dispatch(toggleChecked(task)),
43   };
44 };
45
46 export default connect(null, mapDispatchToProps)(Task);
47 |
```

- Hướng dẫn: actions

```
src > actions > JS authActions.js > ...
1 export const signIn = creds => {
2   return (dispatch, getState, { getFirebase }) => {
3     //console.log(creds);
4     const firebase = getFirebase();
5
6     firebase
7       .auth()
8       .signInWithEmailAndPassword(creds.email, creds.password)
9       .then(() => {
10         var user = firebase.auth().currentUser;
11         console.log(user);
12         console.log("SIGNIN");
13         dispatch({ type: "SIGN_IN" });
14       })
15       .catch(err => {
16         dispatch({ type: "SIGN_IN_ERR" }, err);
17       });
18   };
19 };
20
21 export const signOut = () => {
22   return (dispatch, getState, { getFirebase }) => {
23     const firebase = getFirebase();
24
25     firebase
26       .auth()
27       .signOut()
28       .then(() => {
29         dispatch({ type: "SIGN_OUT" });
30       });
31   };
32 };
33
```

```
src > actions > JS taskActions.js > addTask > <function> > then() callb
1  export const addTask = (task) => {
2    return (dispatch, getState, { getFirestore }) => {
3      const firestore = getFirestore().firestore();
4      const authorId = getState().firebase.auth.uid
5
6      firestore
7        .collection("tasks")
8        .add({
9          ...task,
10         authorId: authorId,
11         date: new Date(),
12       })
13       .then(() => {
14         dispatch({
15           type: "ADD_TASK",
16           task,
17         });
18       })
19       .catch((err) => {
20         dispatch({
21           type: "ADD_TASK_ERR",
22           err,
23         });
24       });
25     });
26   };
27
28   export const removeTask = (task) => {
29     return (dispatch, getState, { getFirestore }) => {
30       const firestore = getFirestore().firestore();
31       firestore
32         .collection("tasks")
33         .doc(task.id)
34         .delete()
35         .then(() => {
36           dispatch({
37             type: "REMOVE_TASK",
38           });
39         })
40         .catch((err) => {
41           dispatch({
42             type: "REMOVE_TASK_ERR",
43             err,
44           });
45         });
46     });
47   };
48 }
```

*** Yêu cầu nộp bài:

SV nén file (hoặc share thư mục google drive) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM.

--- Hết ---