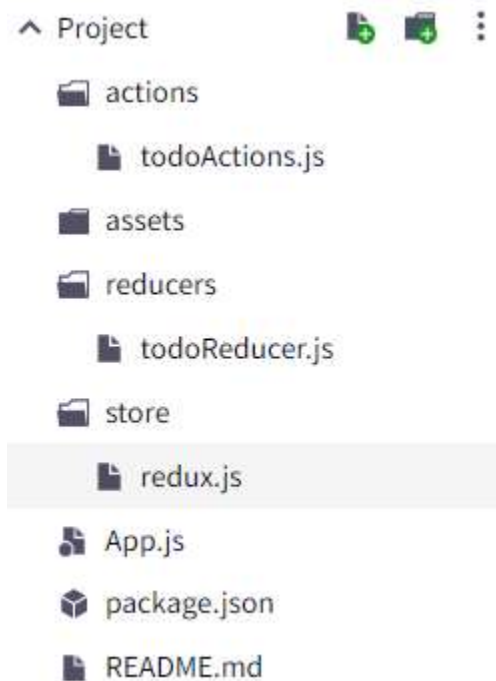


Bài thực hành Tạo ứng dụng TodoApp

- Truy cập <https://snack.expo.dev/>
- Trong thư mục dự án, xóa thư mục **components**, xóa toàn bộ nội dung trong tập tin **App.js** và tạo ra cấu trúc thư mục cùng các tập tin tương ứng như sau:



- Nội dung cho tập tin **todoActions.js**

```
const ADD_TODO = 'ADD_TODO';

const addTodo = (text) => ({
  type: ADD_TODO,
  payload: text,
});

export { ADD_TODO, addTodo };
```

- Nội dung cho tập tin **todoReducer.js**

```
import { ADD_TODO } from '../actions/todoActions';

const initialState = {
  todos: [],
};
```

```

const todoReducer = (state = initialState, action) => {
  switch (action.type) {
    case ADD_TODO:
      return {
        ...state,
        todos: [...state.todos, action.payload],
      };
    default:
      return state;
  }
};

export default todoReducer;

```

- Nội dung **Redux.js**

```

import { createStore, combineReducers } from 'redux';
import todoReducer from '../reducers/todoReducer';

const rootReducer = combineReducers({
  todos: todoReducer,
});

const store = createStore(rootReducer);

export default store;

```

- Nội dung **App.js**

```

import React, { useState } from 'react';
import { View, Text, TextInput, TouchableOpacity } from 'react-native';
import { Provider, connect } from 'react-redux';
import store from './store/redux';
import { addTodo } from './actions/todoActions';

const App = (props) => {
  const [text, setText] = useState('');

  const handleAddTodo = () => {
    props.addTodo(text);
    setText('');
  };
};

```

```

return (
  <View>
    <TextInput
      value={text}
      onChangeText={setText}
      placeholder="Enter a todo"
    />
    <TouchableOpacity onPress={handleAddTodo}>
      <Text>Add Todo</Text>
    </TouchableOpacity>
    {props.todos.map((todo, index) => (
      <Text key={index}>{todo}</Text>
    ))}
  </View>
);
};

const mapStateToProps = (state) => ({
  todos: state.todos.todos,
});

const mapDispatchToProps = {
  addTodo,
};

const ConnectedApp = connect(mapStateToProps, mapDispatchToProps)(App);

export default function Main() {
  return (
    <Provider store={store}>
      <ConnectedApp />
    </Provider>
  );
}

```

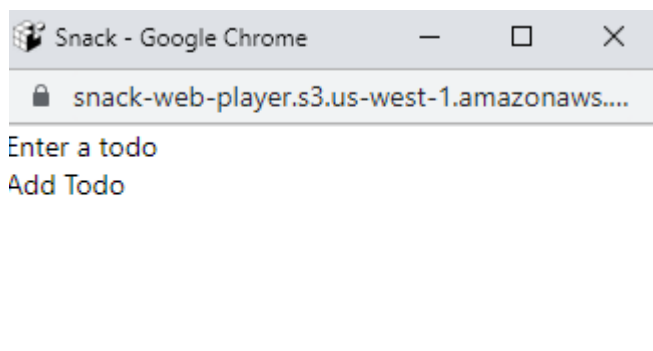
Lúc này hãy nhìn phía dưới màn hình của giao diện Snack Expo sẽ xuất hiện thông báo yêu cầu thêm các gói Redux và React Redux đến Dependencies của ứng dụng



```
21 .....<TouchableOpacity·onPress={handl
22 .....<Text>Add·Todo</Text>
23 .....</TouchableOpacity>
24 .....{props.todos.map((todo,·index)·=
25 .....<Text·key={index}>{todo}</Text
26 .....)}}
27 .....</View>
```

❌ App.js (3:35) 'react-redux' is not defined in dependencies. [Add dependency](#)

Nhấn chuột đến liên kết **Add dependency** cho đến khi thêm đầy đủ các gói cần thiết. Lúc này ứng dụng trông như sau:



Nhập một công việc vào mục **Enter a todo** và nhấn nút **Add Todo**

