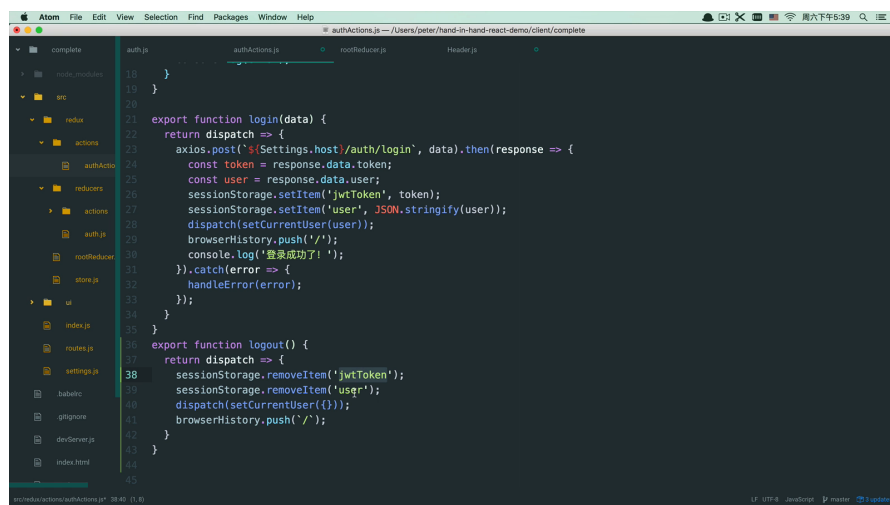


[首页](#) [目录](#)[退出](#)

引入 Material-UI 组件  
库美化页面

借助 Radium 工具  
集，编写导航栏的样  
式

通过 React Router 添  
加前端页面的路由功  
能

## 前端实现用户退出登录功能

### 定义 Action 创建函数 logout

打开 `src/redux/actions/authActions.js` 文件，定  
义一个 `logout` 方法：

```
export function logout() {  
  return dispatch => {  
    sessionStorage.removeItem('jwtToken');  
    sessionStorage.removeItem('user');  
    dispatch(setCurrentUser({}));  
    browserHistory.push('/');  
  }  
}
```

## Header 组件调用 logout 方法

打开 `src/shared/Header.js` 文件，导入 Action 创建函数 `logout`：

```
import { logout } from '../../redux/actions/;
```

然后，借助 `connect` 方法，把 `logout` 方法注入到 Header 组件：

```
export default connect(mapStateToProps, { log
```

这样，我们就能在 Header 组件中调用 `logout` 方法，分发 action，更新 store 中的 state 了。

接下来，给 退出 登录链接添加一个 `onClick` 事件，其对应 `logout` 事件处理函数：

```
<Link to="/" style={styles.nav} onClick={this
```

定义 `logout` 事件处理函数，调用 Redux Action 创建函数 `logout`：

```
logout(e) {  
  e.preventDefault();  
  this.props.logout();  
}
```

## 解决刷新页面退出登录的问题

打开 `src/routes.js` 文件，导入 `setCurrentUser` action:

```
import { setCurrentUser } from './redux/actions'
```

判断 `sessionStorage` 中是否保存了 `jwtToken` 条目，若存在，则获取在 `sessionStorage` 中保存的用户信息 `user`，然后直接调用 `Redux store` 的 `dispatch` 接口，分发由 `setCurrentUser` 函数返回的 `Redux Action`，触发 `Redux store` 中的 `auth state` 更新

```
if (sessionStorage.jwtToken) {  
  const user = JSON.parse(sessionStorage.user);  
  store.dispatch(setCurrentUser(user));  
}
```

欢迎添加 Peter 的微信: happypeter1983

冀ICP备15007992号-3