# React-Redux-Firebase-App

A project management tool using ReactJS/React-Redux JavaScript using thunk as middleware and moment JavaScript for dates. It is complete with Google’s Firebase that provides Authentication, Cloud Functions, hosting, Firestore).

Firebase Firestore as a real time db.

Firebase cloud functions notifications.

Firebase authentication for authentication flow

Firebase deployment

-------------------------------------------------------------------------

Structure of the WebApp:

Create-react-app

App

|

NAV BAR

| |

Signed in Links Signed out Links

{Either one of these will be shown depending on the authentication of the user}

App component branches out to:

1. Dashboard Component : a)Project List b)Notifications

2. Project with its specific ID that shows that project

3. Create Project ID - will have a form

4. Sign in

5. Sign out

**React-Redux:** React-Redux is a package that glues react and redux together. It has the Provider component and the connect component.

**Redux-Thunk:** Middleware for redux. It returns a function instead of an action (that happens in Redux) and write it to action-creators.

**React-Router:** React router is the standard routing library for React. It makes sure UI is in Sync with URL.

The previous way was doing static routing. This is a dynamic way of doing it as it does most of its work on the client side.

When a component does not have state, it is going to can have functional components.

A **functional component** is just a plain JavaScript function which accepts props as an argument and returns a React element. A class component requires you to extend from React Component and create a render function which returns a React element.

Firestore: is the Realtime - NoSQL database.

Cloud Functions for Firebase let you automatically run backend code in response to events triggered by Firebase features and HTTPS requests.

React-redux-firebase: Provides bindings to the firebase service as a whole

Redux-firestore: Provides redux bindings for Firestore databases

**Redux architecture consists of :**

1.Service

2.Action and Action Creator

3.Reducer

4.Selector

5.Container

What props are to React, Store is to Redux.

We create multiple reducers for different parts of our app that each handle a small set of actions. Then combine those reducers to one root-reducer and pass it into the store.

In this case "index.js" has the store. Auth and Project and two reducers and they are combined in the root reducer.

"combineReducers" is a function that combines reducers.

We use think of **thunk** as middleware to pass dynamic data from a database.

We can't pass data through to the reducer or to the component area.

Redux-thunk will allow us to perform asynchronous tasks inside of our action creators.

Usually once an action is dispatched, it shall create an action creator and then return it. But with thunk we return a function instead. In this function we can halt the dispatch, perform async request(like grabbing data or update data) and resume dispatch.

Collections are documents. They are usually a key value pair.

In this project we are going to have three types of collections:

1. Projects collections

* Title
* Content
* authorFirstName
* authorLastName
* authorID
* timestamp

1. Users collections

* Info about the app users

1. Notifications collections

* Info about notifications

**<Read what the spread operator is>**

**<Read what a Provider does>**

**<How firebase and Redux code works together on #18>**

**<Read what a state object in a reducer is>**

**<What does higher order component mean>**

**<Number 12 has a lot of explanation.>**

**<Number 13 has different methods to have done the project and why he didn’t do it>**

**<#20 explains how we get project details>**

**<#21 explains how we structure users collection>**

**<#28 explains how we protect the content from people who haven’t signed up or signed in>**

**Packages install:**

npx install create-react app

npm install redux

npm install react-redux

npm install redux-thunk

npm install react-redux-firebase

npm install redux-firestore

npm install moment

npm install -g firebase -tools

firebase init

firebase login

npm run build

firebase deploy