React Project

My project is completely build using React JS and Redux. This whole project has multiple projects, each one of them is significant to depict UI on the browser. So, I have explained it in the video I made.

The application I developed is completely present in ‘src’ folder.

So, starting from the Top:

1. Components:

This folder contains all the component we mainly used to render on to the screen. These objects are called by the scripts in containers, which imports multiple components scripts.

1. Burger:
   1. Build Controls: This field contains the add ingredients and remove ingredients. Which are + and – placed along with every ingredient like cheese and bacon, to add and remove them from the burger.
   2. Burger Ingredients: This sub-folder is used to access the ingredients between breads which we are adding and removing.
   3. Order summary: it will contain all the information about order like what are the ingredients and total price of the burger, and confirms if you want to proceed with the project or not
2. Logo: This folder contains the burger logo which we can show on the top-left corner part of the page
3. Navigation: This folder contains all the UI developed to show on the navigation bar.
   1. Navigation Items: These are the links which we specified in the navigation bar, there three navigation item has NavLink Object it returns, this NavLink is built in object of ‘react-router-dom’ library.
   2. Side-Bar: This field is shown on the screen when width is less that 500px. For that to happen I used media queries in CSS, which are mainly used to render some CSS content based on some condition satisfaction.
   3. Toolbar: It is the navigation bar itself.
4. Order: This whole component is showing the check-out page of the website
   * 1. Checkout-Summary: This contain Burger you ordered by importing the, burger component which was used before, then it also contains two buttons.
        1. First button is continuing to proceed with order and forward you to fill out your details.
        2. Second Button is go back to Burger Building page and create new customized burger.
     2. Order: this is the java script file that calculates price of the burger and all their ingredients.
5. User Interface (UI):
   * + 1. Backdrop: this field creates a black aura on the back if we press on order button and the order summary pops up on to the screen
       2. Button: This is the component which will have templates for different kind of buttons. So instead of using the bootstrap we import button from this container and specify the type we want, if we pass parameter success then the green button template is generated and if we pass danger in that same parameter then the red color button is generated,
       3. Input: This field is used to generate input fields. Like text-input, text arear, dropdown list based on the string parameter we pass to the component when we call it from a container.
       4. Modal: this field which appears and disappears when we press the “Order Now” on the main page. It used as a background to render our order summary.
       5. Spinner: This is the small component rendering on the screen when there is delay between retrieving or sending the information to the server.
6. Containers:
7. Burger Builder:

This is the first page that renders on to our site. This container renders many components from the above folder. This page renders the controls to increase and decrease the number of ingredient inside the burger, finally before proceeding to the checkout page it renders the order summary

* 1. Burger
  2. Build-Controls
  3. Modal
  4. Order summary
  5. Spinner

ii. Checkout:

This page contains the burger you ordered and form to enter the personal information and address from the customer and allows us to store the order details along with user details to the firebase database.

1. Contact-Data:

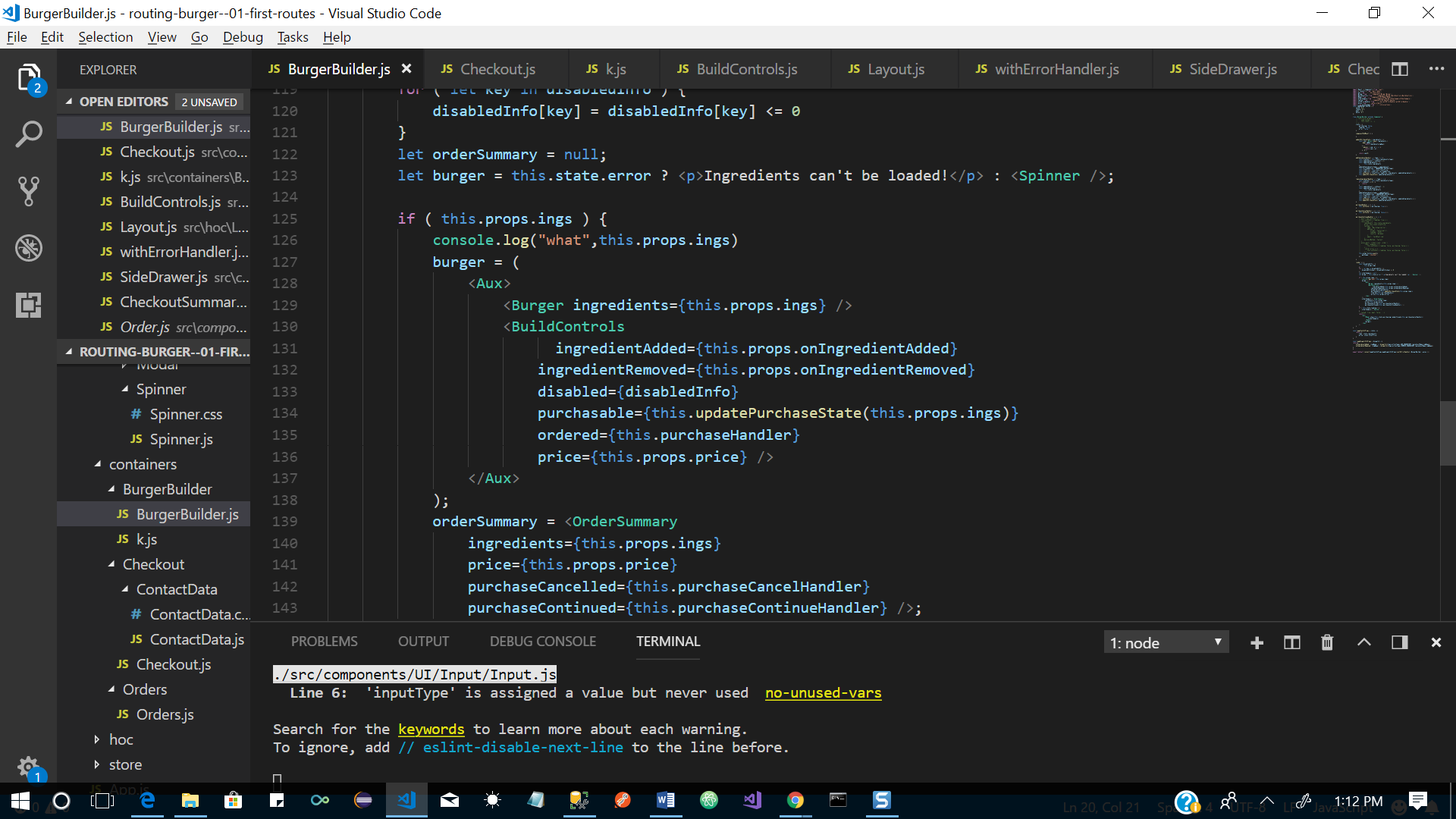
It contains the form which we create inside the checkout page, the input fields inside this page are generated using the input component in the component folder.

iii. Orders:

This is a page where we render our orders, by retrieving them from the firebase database.

1. Higher Order Components:

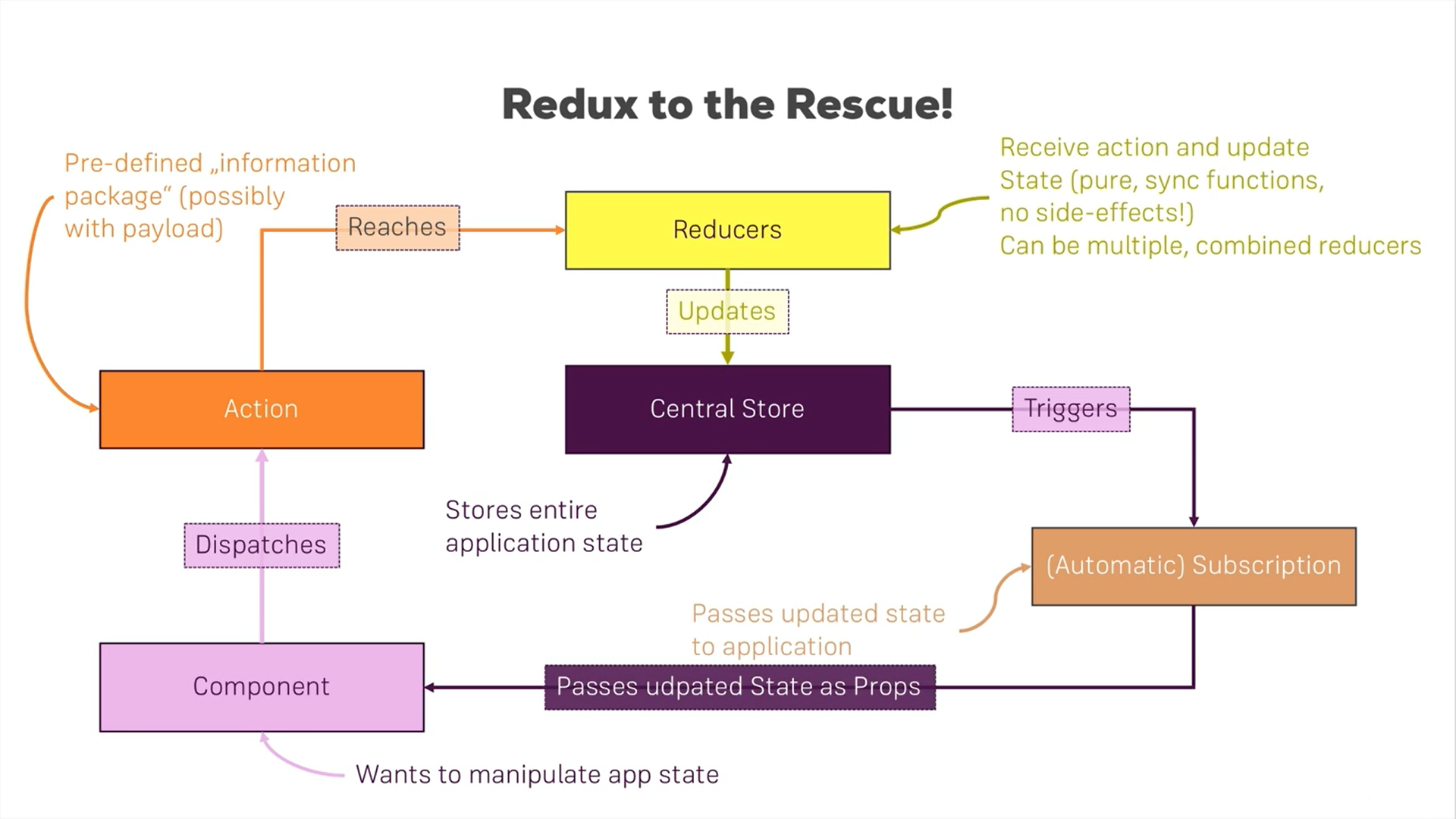
These higher order components are used to wrap normal components in them, because normally in a react only component can be specified or called inside one JS file, if you want to call another component inside the same Java-script file then you have to wrap both of them together with a higher order component.



As you can see in the above code we can’t specify Burger and Burger controls side to side we have wrap them in a higher order component. So, we imported Aux and wrapped both the component in it.

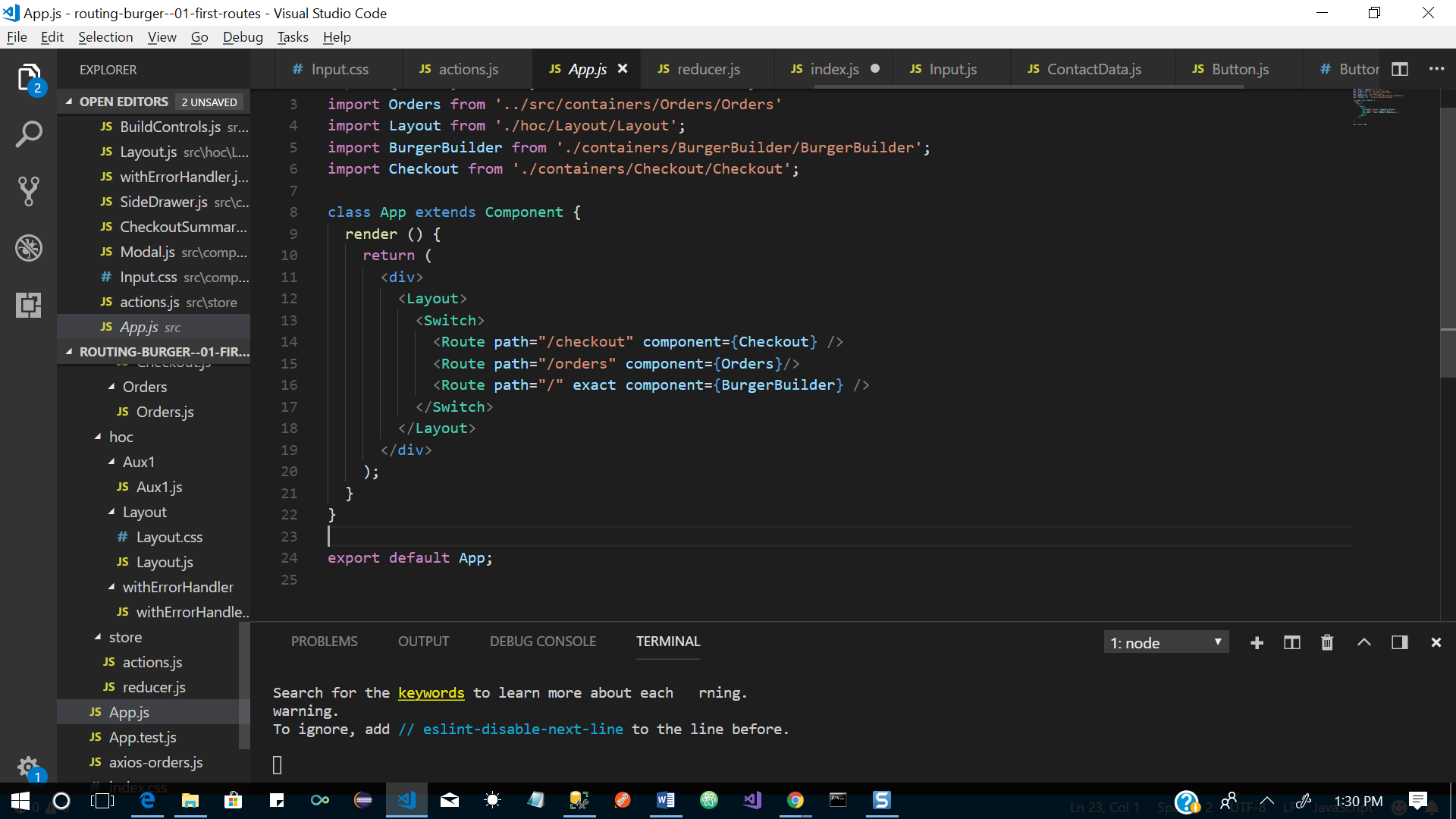
* 1. Aux1: It is the higher order which is specifically use for HOC purpose in this project.
  2. Layout: This contains the toolbar and navigation Bar components like Tool bar and sidebar which are displayed based on the width of the screen.
  3. With-Error-Handler: This is the component which is used to catch the errors from the components, it wrapped around and render the error message as an alert. It uses axios libraries interceptors to catch the request and response errors sent to the server.

1. Store: This is redux component, which handles the state of the project, this state holds the global information of the project which cannot be manipulated by any component directly, to manipulate a state element from a component we must connect it to store, dispatch some actions which will be send to repository and repository changes state according to the type of action it received, then the state will be updated in all the components at once.



* 1. Actions: This is the file which contains all the action names stored in a const variable, because we do not want to call our action by a string every time(There may be a consequence of misspelling a action, but if we store it in a variable then misspelling of a variable name will be caught by the compiler).
  2. Reducer: This is used to manipulate the state based on the kind of action send to it.

App: App.js file contains Route to all the containers or pages in our application.



Axios-orders: This is the file which is connects our app to the base URL of my firebase database.

import axios from 'axios';

const instance = axios.create({

baseURL: 'https://react-my-burger-6796e.firebaseio.com/'

});

export default instance;