**Pre-requisites:-**

**1.** Node should be installed on ur system jus google it and install.

**2.** Angular should be installed using the command

**npm install –g @angular/cli**

(3,4,5 th steps used while creating the project)

**3.** Fake json server should be installed using

**npm install -g json-server**

**4.** How to start json server ?

**json-server - -watch db.jon** (type this command in **cmd**)

**5.** Bootstrap setup

**6.** To view in browser type **ng serve** in vs code cmd.(**ng serve -o** - it’ll automatically take u to browser)

**Steps to create project:-**

**1.** How to create new project ?

**ng new projectname**

**2.** Install json server inside the project **(npm install -g json-server)[**it’ll have all the collections of Contacts which will be created later**]**

**3.** After the installation let’s go to Bootstrap setup.Jus go to Bootstrap website.From here need to copy two things which is the link of CSS and js.Copied links should be pasted inside the index.html file (paste css link inside the head tag and js link inside the body tag).

**Implementation:-**

**1.** Create dashboard using the command

**ng g c dashboard** which means(ng generate component dashboard)

**2.** Then go to dashboard.ts copy the selector and paste inside the app comp.html (check whether it works or not)

**3.** Create nav bar and table inside the dash.html.Inside the th give the column names & inside the td give the values.Next style the btns using bootstrap classes.

**4.** By clicking on Add Contact,, modal should open.Copy the modal code from bootstrap and paste below the table.Copy the button trigger modal tags and paste on our Add Contact.

**5.** Then add the forms.

**6.** import form builder & form group inside the dash.ts (it ll bind the input values).Inside the class initialize form builder.Values should be passed inside the constructor.Next add(inside the ngOnInit) all the formcontrols which we have in the particular form.

**7.** In dash.html inside the form tag add form group & form value which will bind the data from ts file.Each i/p should have formControlNames.We’ll get some error so that we have to import reactiveformmodule inside the app mod.ts file.

**8.** Now we need to post all data into json server.So that install json server using command which i’ve written above.Run the server using localhost.Inside the posts u have post our form data.

**9.** Create four Api’s for get,post,delete.For doing api calls create api service.

1t create folder in the name shared,inside the shared folder create **ng g s shared/api**(s means service).Delete the spec file.

**Api calls:-**

**1.**  Inside the api file import httpclient.In app mod.ts file we should have httpclientmodule.

**2.** Next inject http clients from constructorin api file.

Create method for posting data(postContact)and return & post the data & paste the link where u ve to post it.Using pipe operator it will map where the method is called.import map in api file.

As like **post** create methods for remaining three…

Create contact dash model.ts.Inside this export DashboardModel.then initialize id name with data type.

Let’s create obj in dash comp.ts to pass obj to the server.By using the object,pass the values and post the data.Subscribe it & console the value and end with alert msg.Else it’ll throw an error.Copy the method(postDashboardetails) and put in add button in html so that it will bind to it.When we press the click button it should trigger the postDashboarddetails method,it will take all tha dashboard obj and post in the server.

Once alert is successful the form has to reset so that u have to call the formValue.After submitting the form it has to close it ryt.So that create id near close button in html.then refer that id in ts file using ref.

**Now we have to call getmethod:-**

Datas entered in the page are static.Create method for get.We have to save the response somewhere so that create data property after obj creation line.Use ngFor to loop with tables.Then cl it in ngInit.Once filling it has to show the data in dashboard page so cl It in postmethod.

**Next we have to upd and del the record:-**

Create method for del.Method should be called in html del btn.

When we click on edit btn the form should open up.cl the methods correctly.Add show add and show upd value.Using ngIf we can hide the buttons.

**LOGIN & SIGNUP:-**

Click the project terminal and create login component using the command

**ng g c login**

Do the same for signup.Add routing to our app.Routing is used to navigate from one comp to another component.In app com.html add router outlet.Inside the app routing mod.ts file add path i.e which path has to load.

If our page is blank then add path .Here I have added login,which will redirect to login.

Then design the loginpage & signup page using bootsrap

Use routerLink to move to another page.

Then we have to configure the json server.

Jus run the json server.Create ur own array in db.json for signupvalues.

Import fromgroup & builder and create a public property.Inject the formbuilder to constructor.

Inside the ngOnInit initialize the values.Inside the html add formgroup.Write formcontrolname for all.Call ngsubmit to submit the values using signup method.

Inject the httpclient inside the constructor.Create signup in ts file.

Do the same for login page.