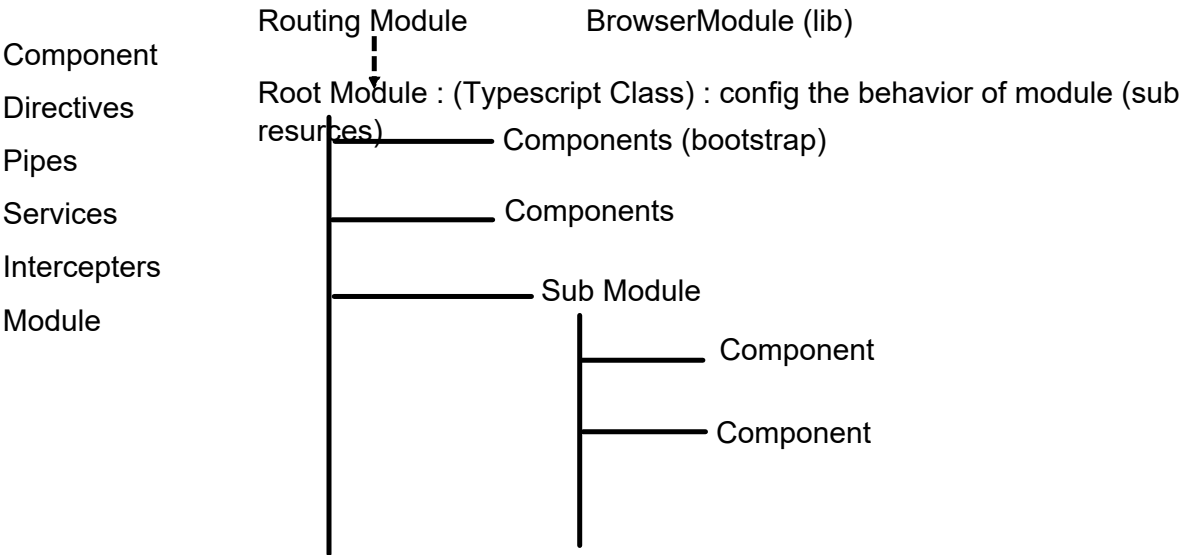


```
> npm install -g <api-name> --save  
(install the api and add an entry into package.json)  
  
# add new entry in package.json  
<api-name> : <version>  
> npm install  
# finds package.json  
(download and install dependencies listed in the file)
```


Javascript Testing framework : JASMINE (available)

Test Runner : Karma (available)



Components

4 files

1. html : UI (optional)
2. ts : Typescript classes
3. css : Custom Stylesheet (optional)
4. spec.ts : unit test code (optional)

New Component Creation

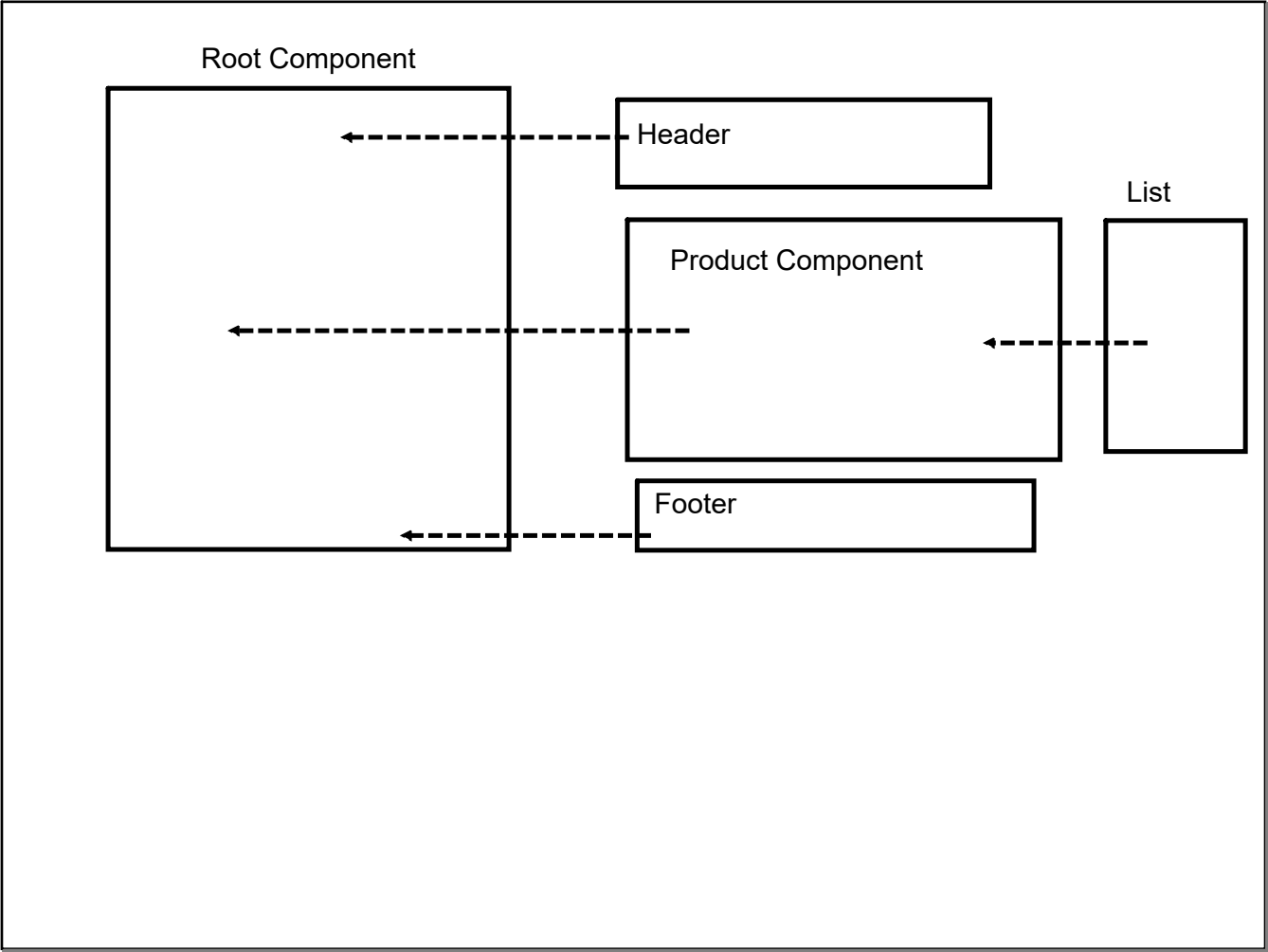
> ng generate component <comp-name>

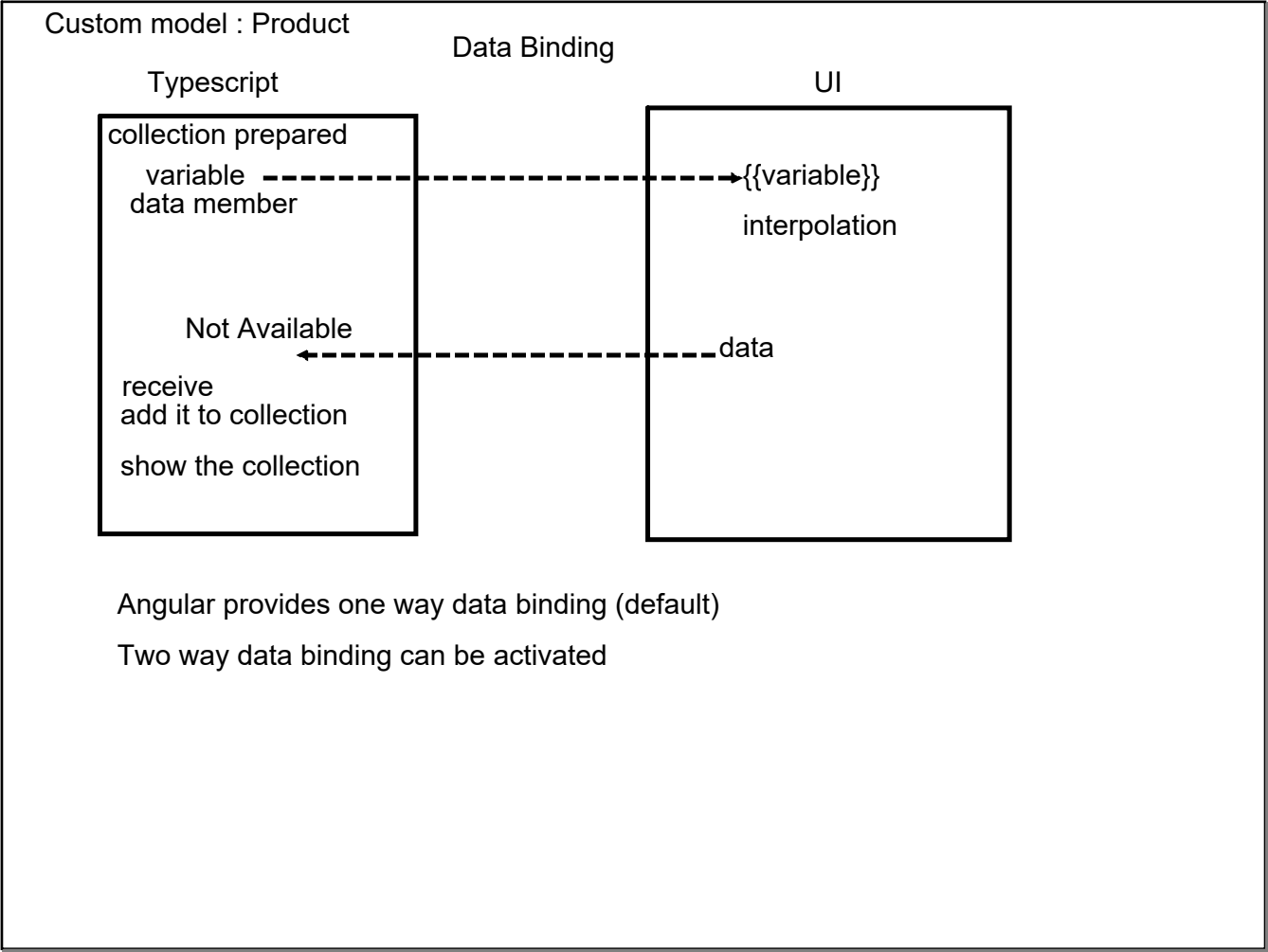
> ng g c <comp-name>

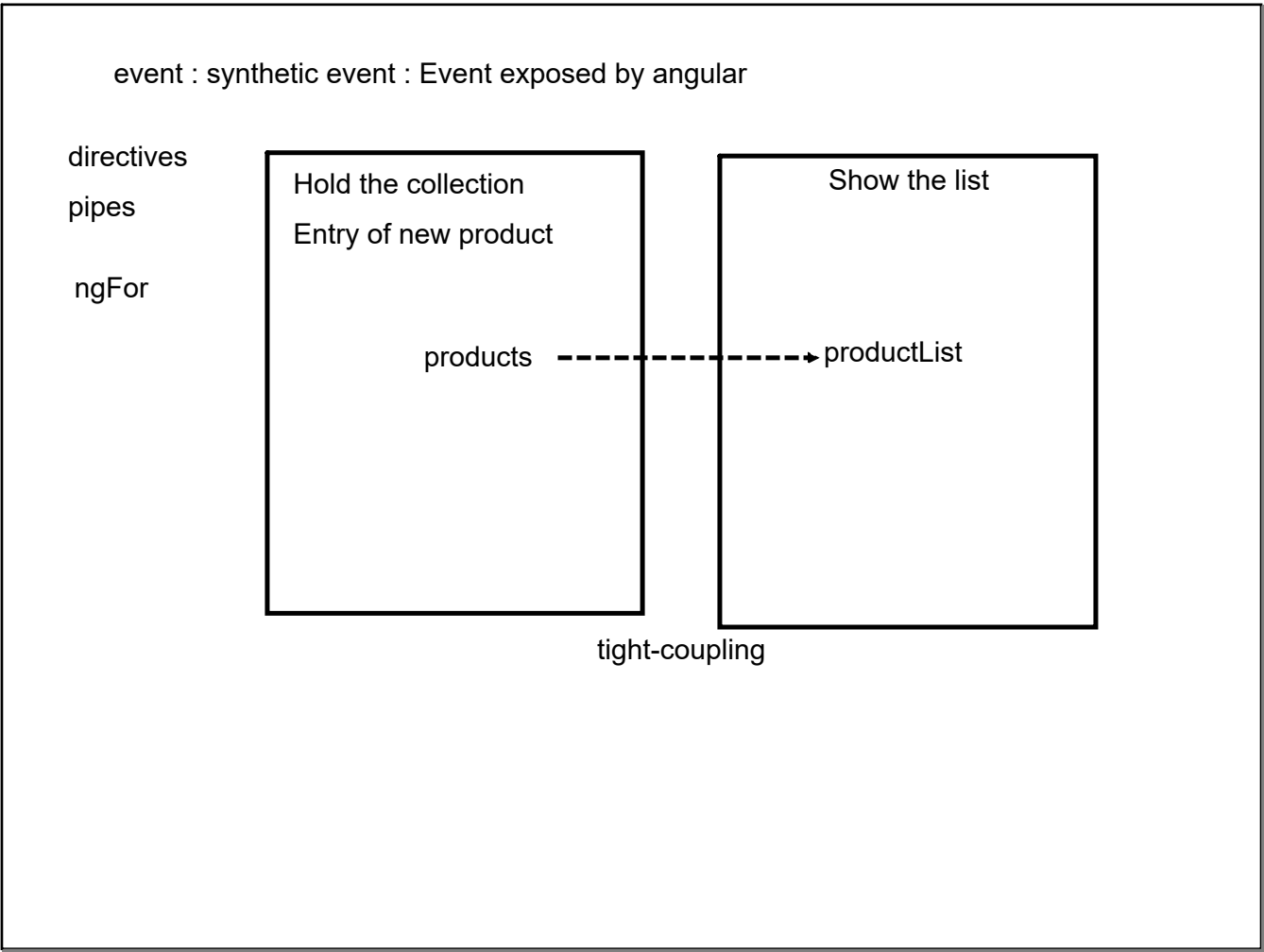
Entry Form to add new product

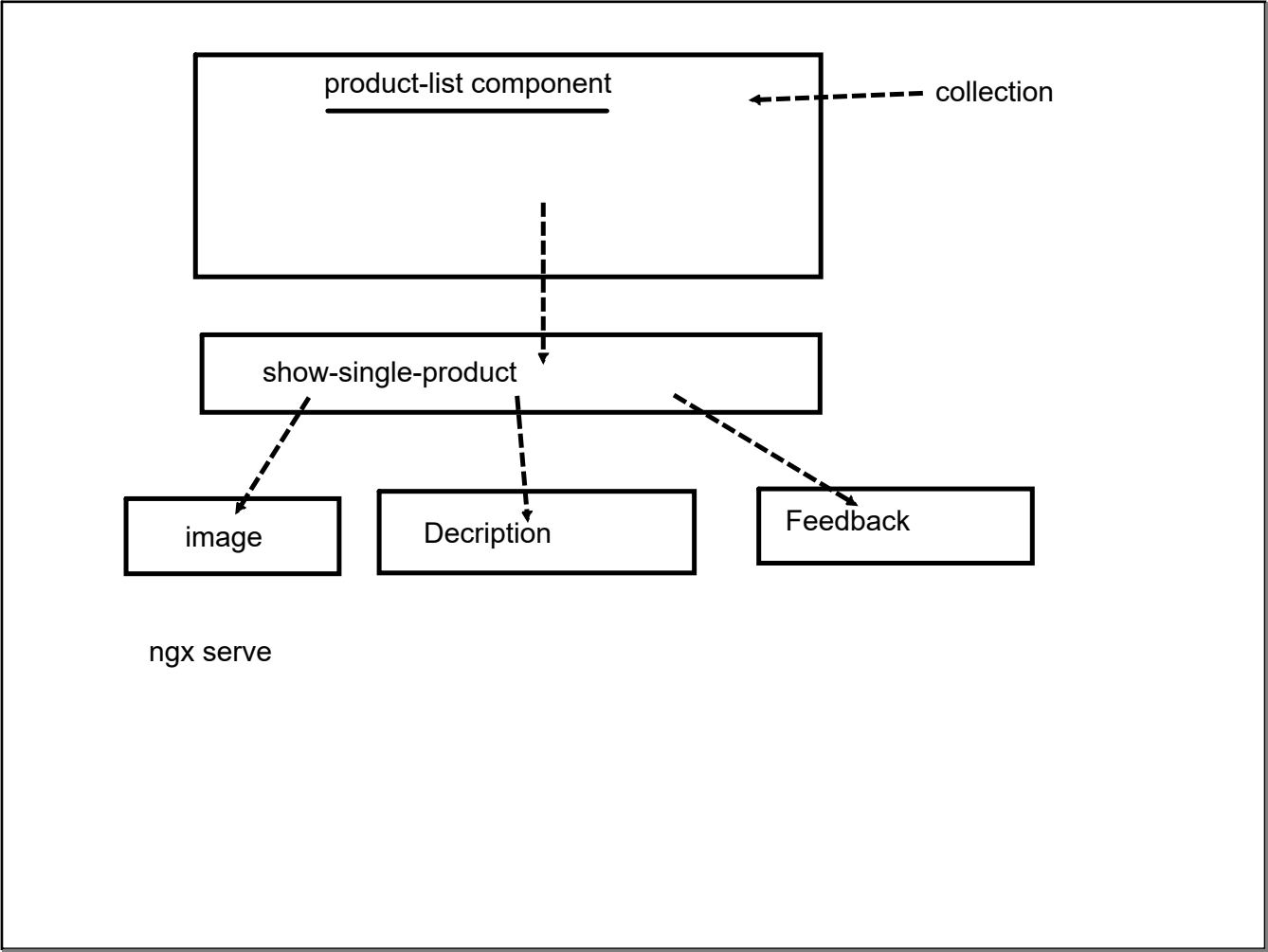
Listing of all products

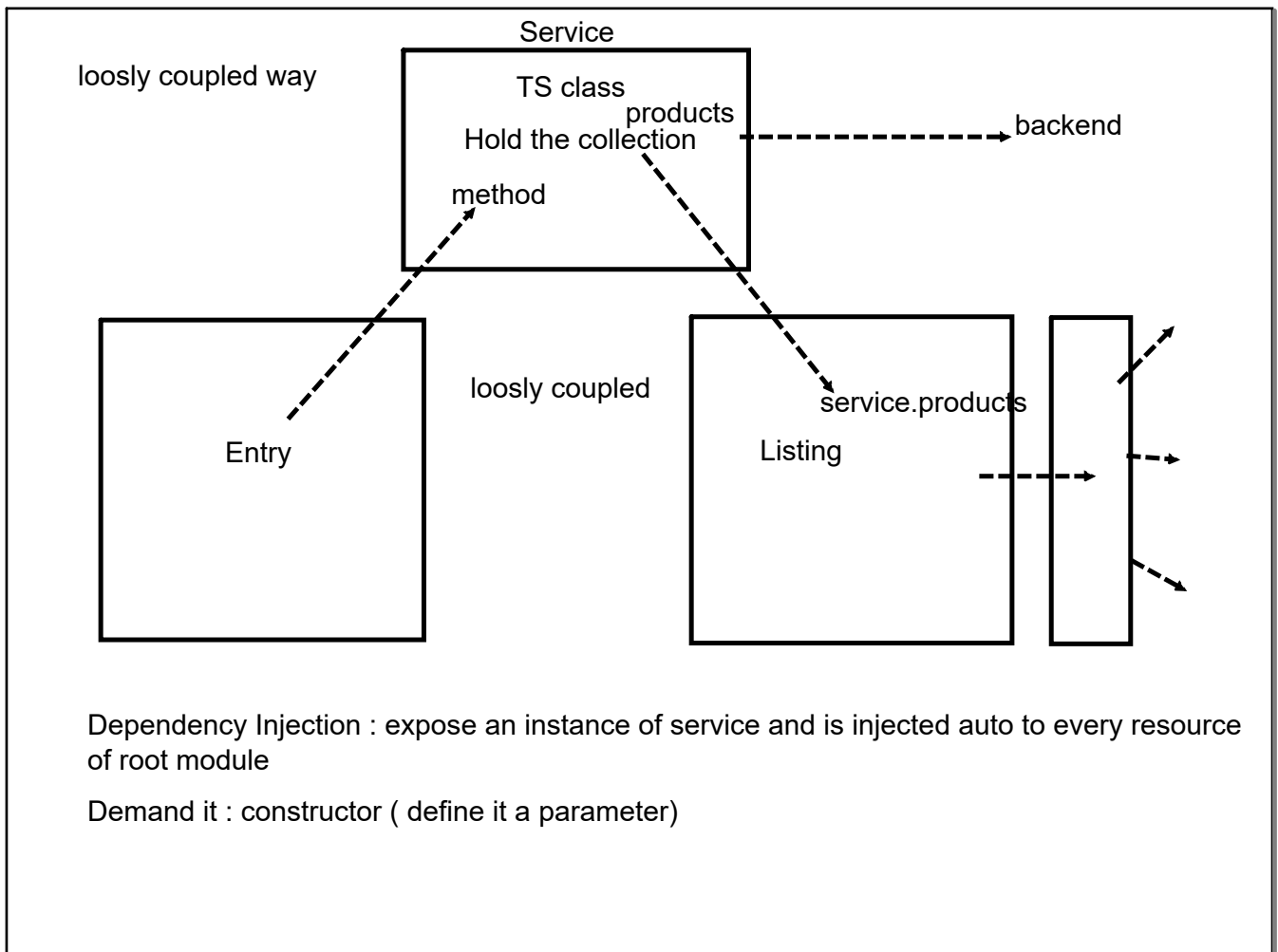
Single Component
Two Component
Three Component

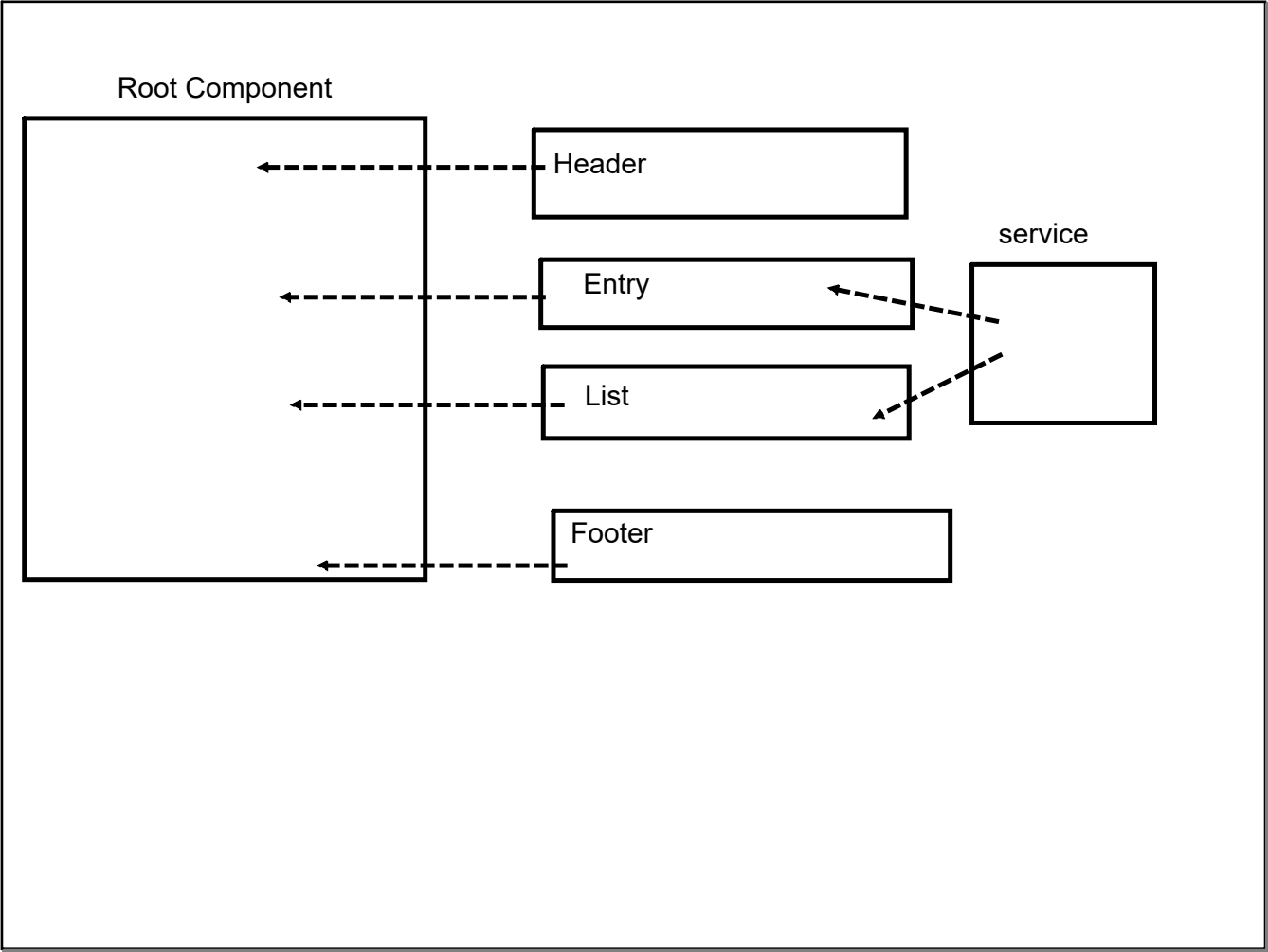












Directive : Structural: DOM Tree

ngFor

ngIf

ngSwitch

based on condition : allows to show or hide some content

Presentation
directive

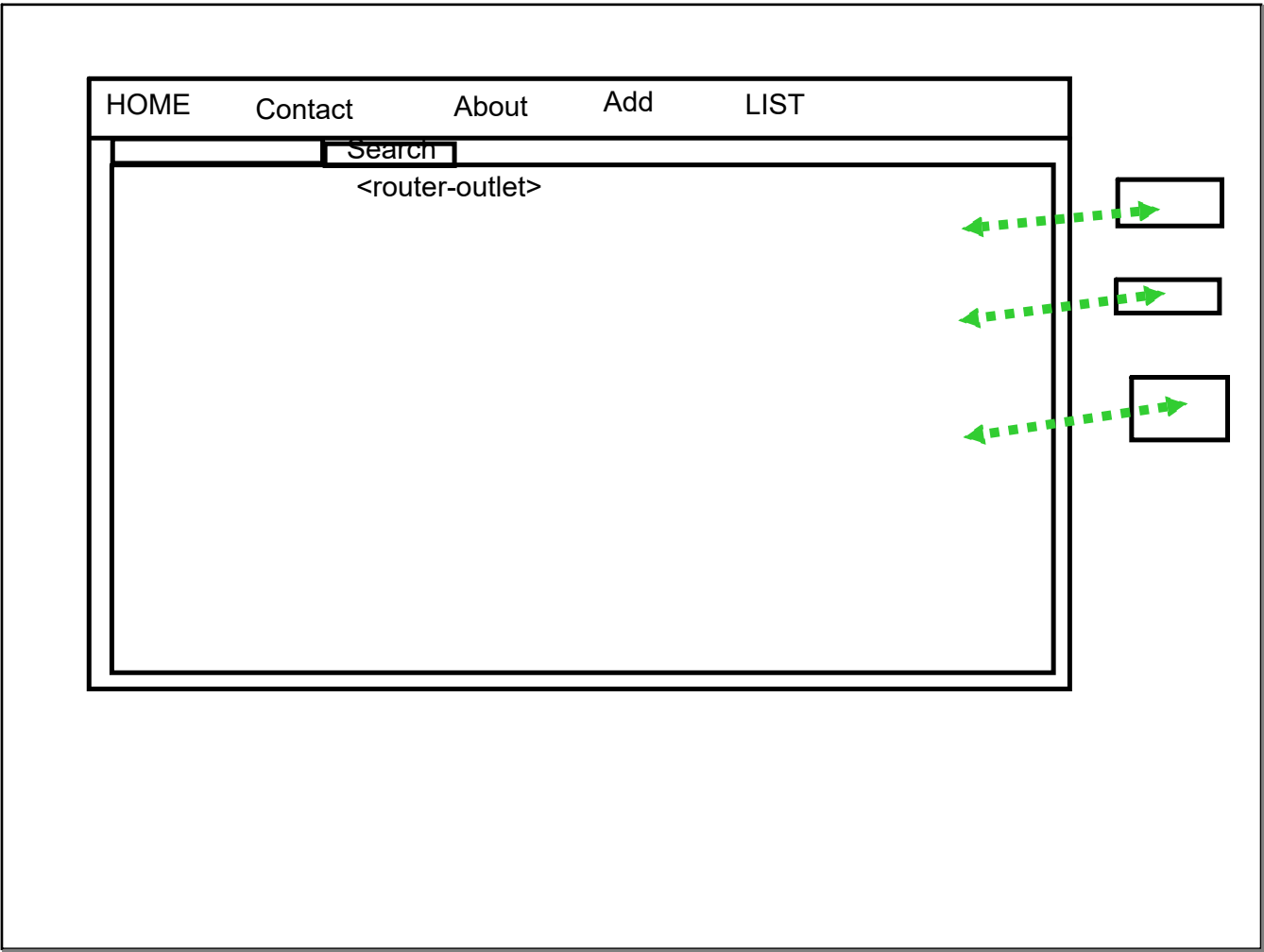
[ngStyle]

[ngClass]

Routing

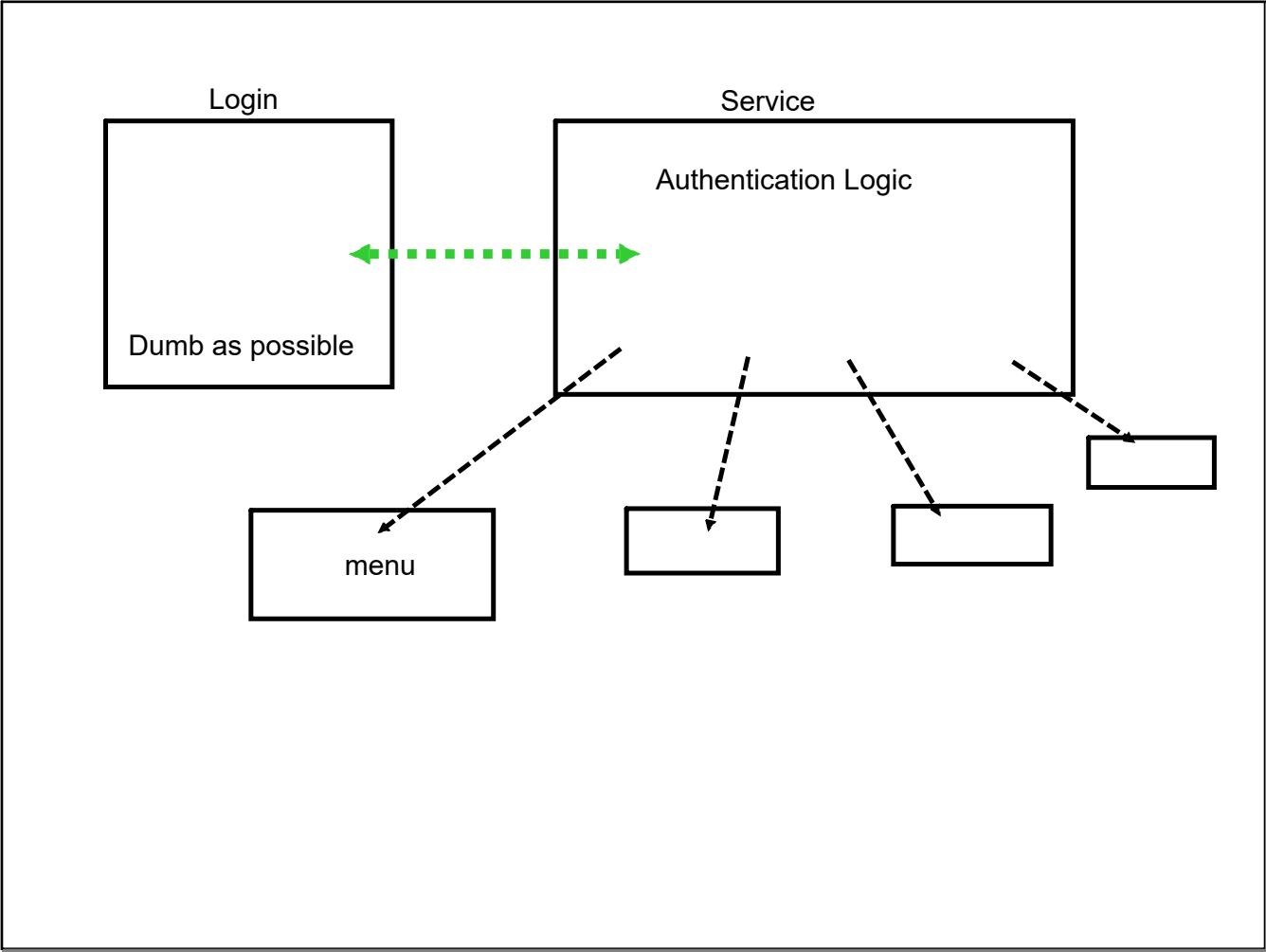
Forms

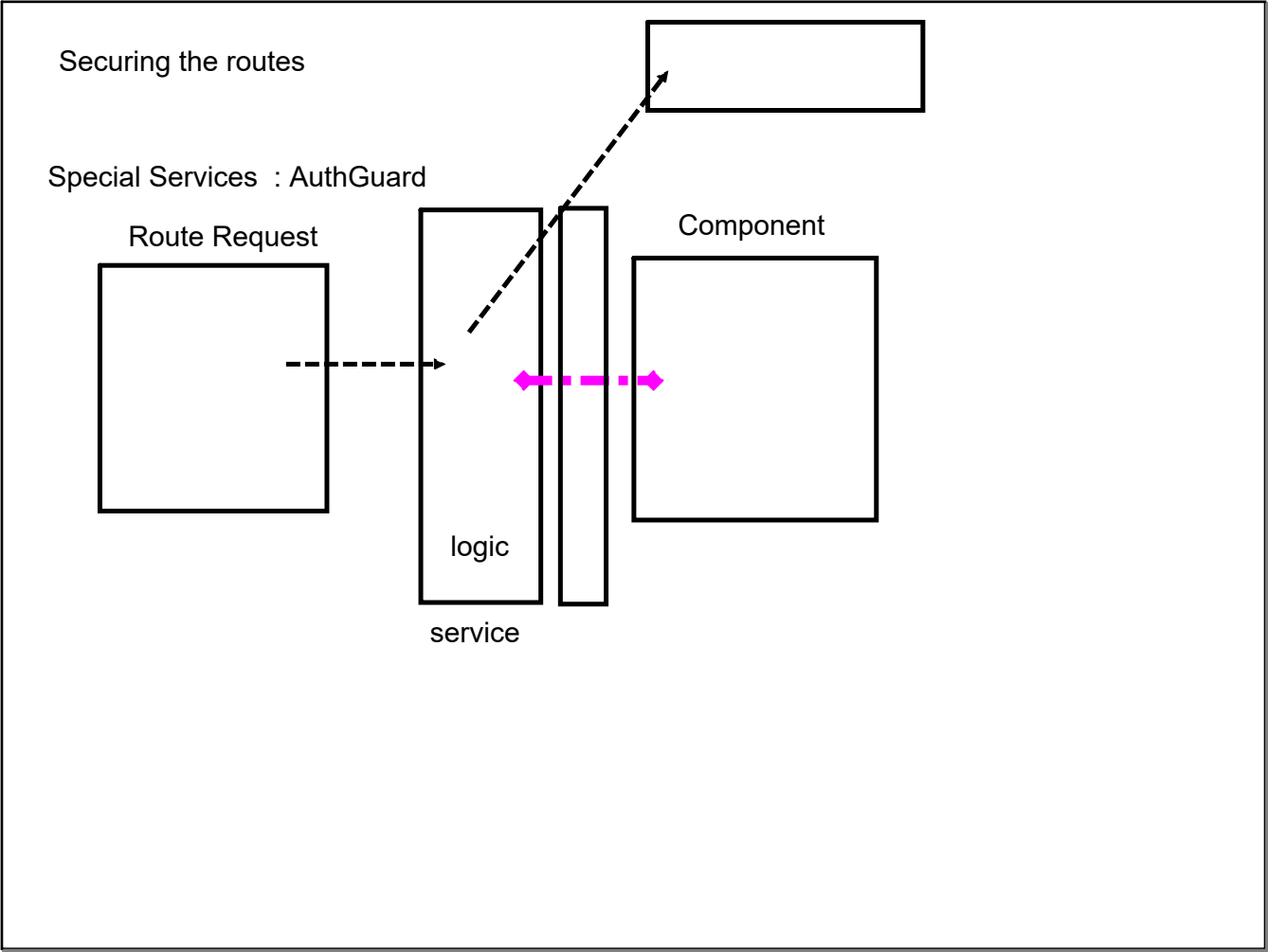
HTTP Client (backend)

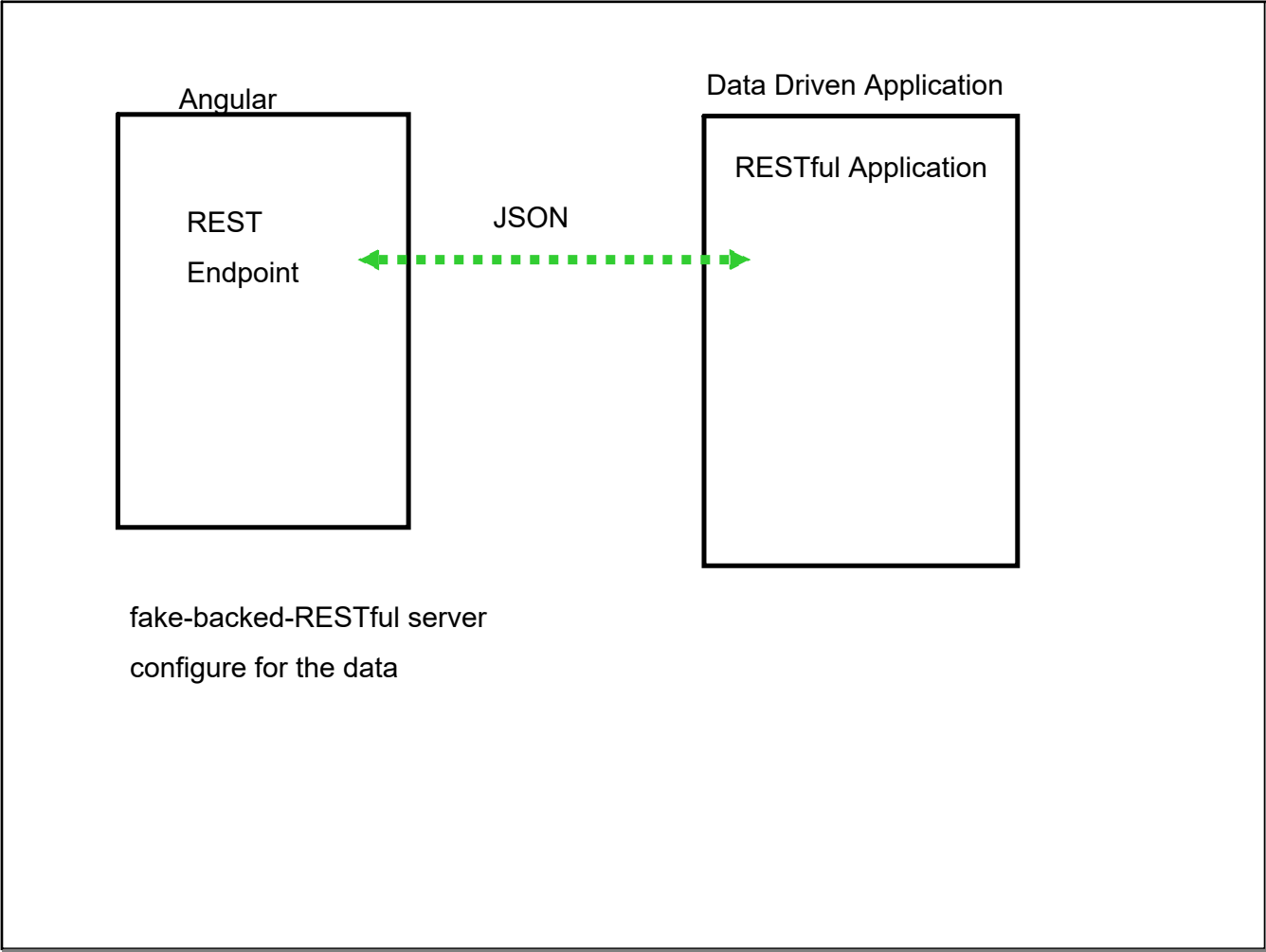


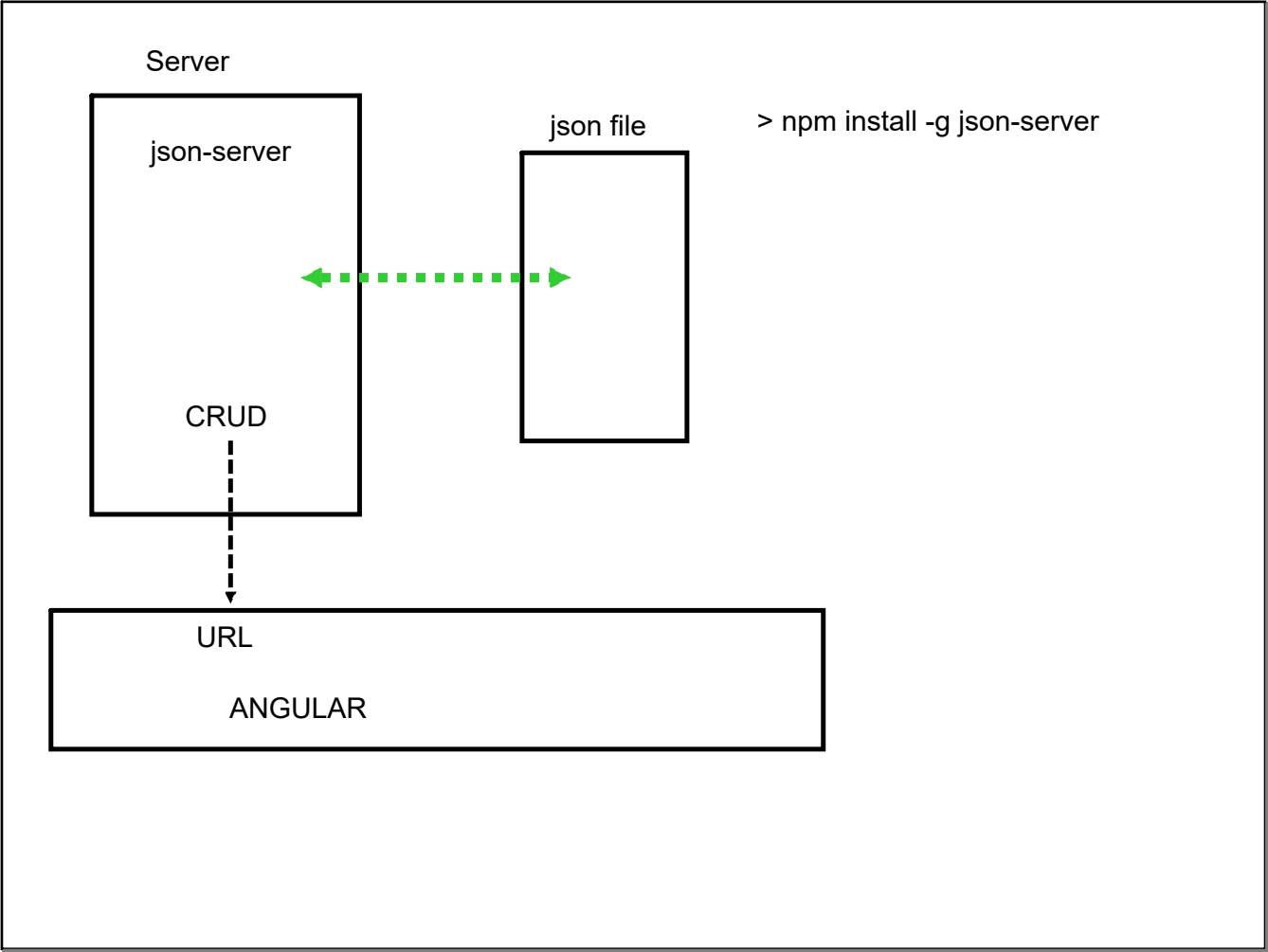
Programmatic Navigation

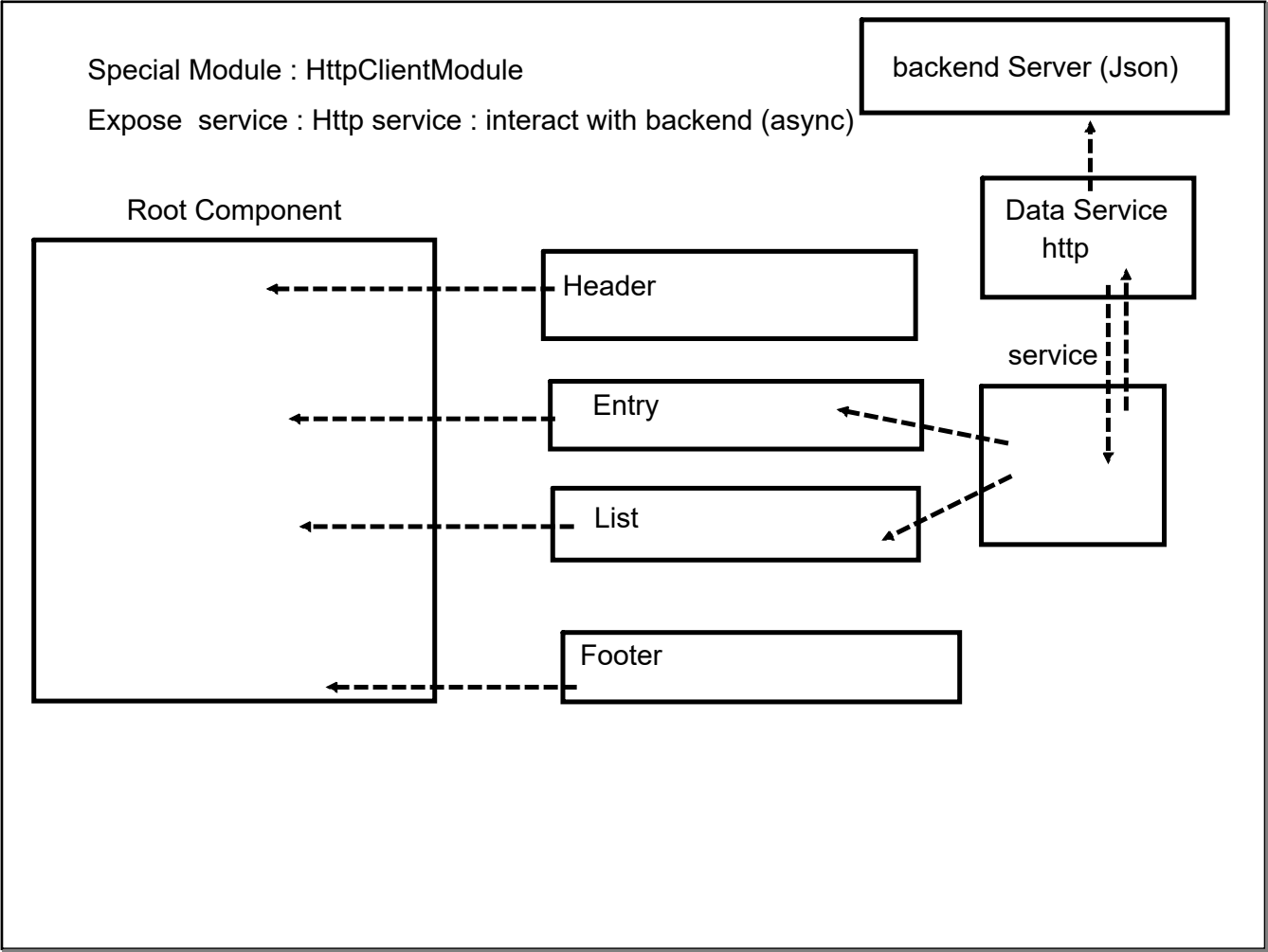
[HOME](#)[ABOUT](#)[CONTACT](#)[LOGIN](#)[HOME](#)[ABOUT](#)[CONTACT](#)[ADD](#)[LIST](#)[LOGOUT](#)











Lifecycle Hooks of Angular Component

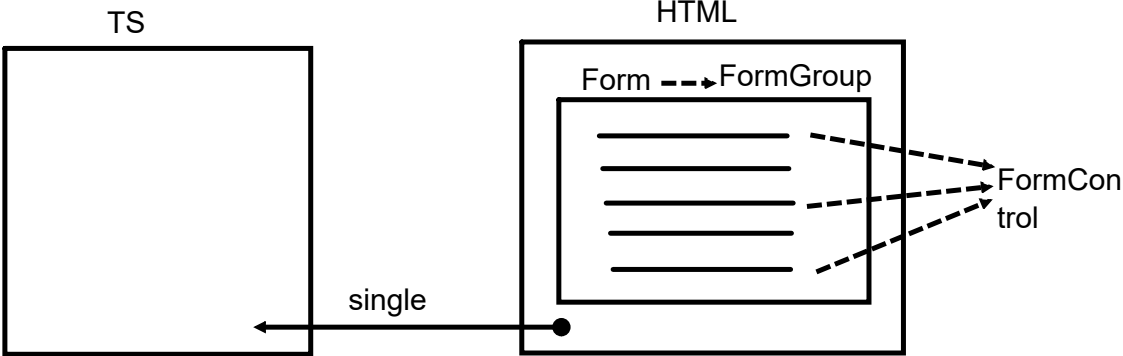
```
this.http.put(API_URL + "/" + id, product);  
this.http.delete(API_URL + "/" + id);  
this.http.get(API_URL + "/" + id);
```



Forms :
 FormControl : represents individual entry field inside the form
 FormGroup : Represents the form (container)

Template Forms

Reactive Forms



Add FormsModule

minlength="5"

max..

min="100"

max=""

pattern="[A-Za-z]+"

