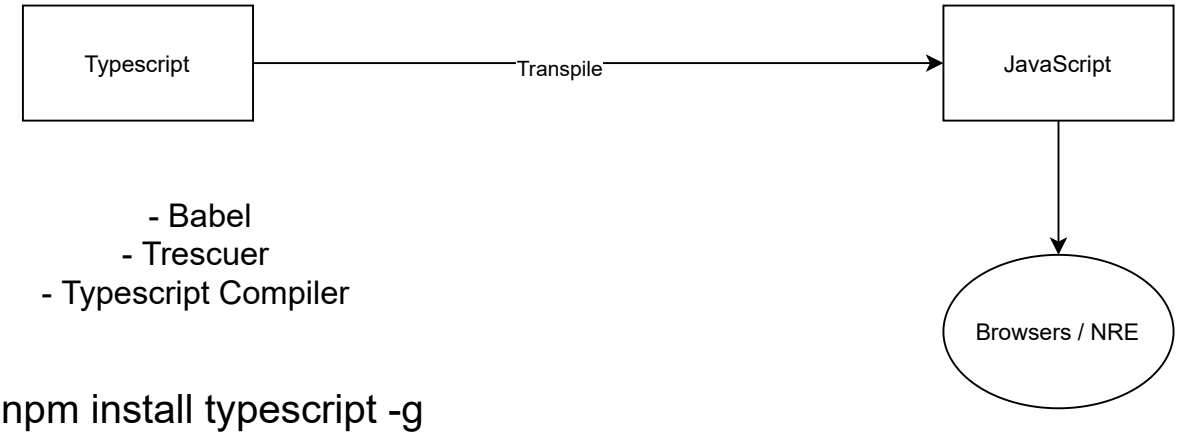
JS Libraries and Frameworks-

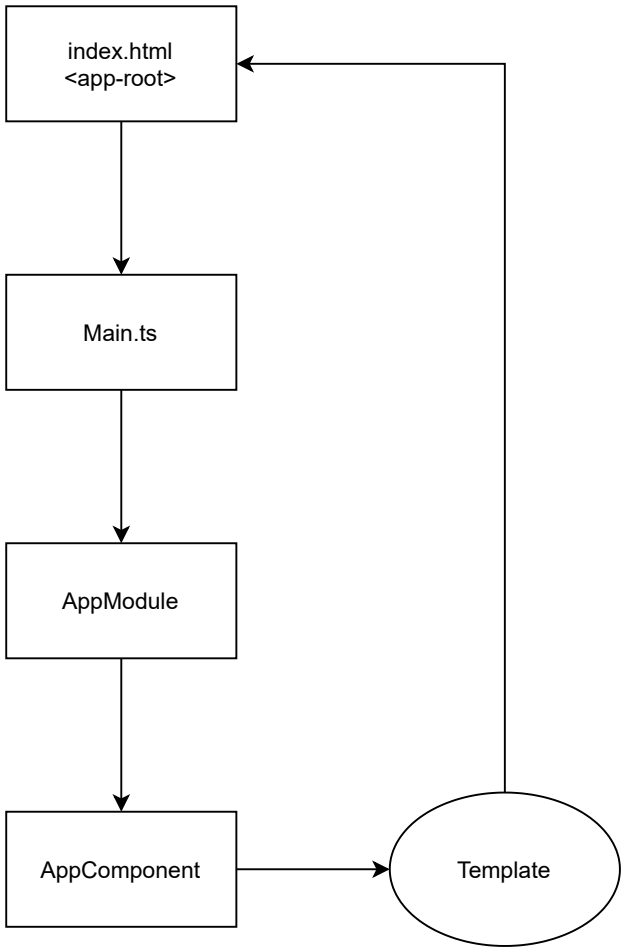
- jQuery : DOM manipulation and ajax
- React - model is changed very frequently. Component based architecture
- KnockoutJS - MVVM- 2 way data binding
- BackboneJS - MVC at client end
- PolymerJS - Custom Components
- EmberJS - Framework, Frequent internal API Changes
- NodeJS - platform; NRE - Server-side script
- ExpressJS - Web app framework for Node App
- VueJS - Library
- Angular : DOM Manipulation, 2 way data binding, SPA, template, MVC, Component etc,

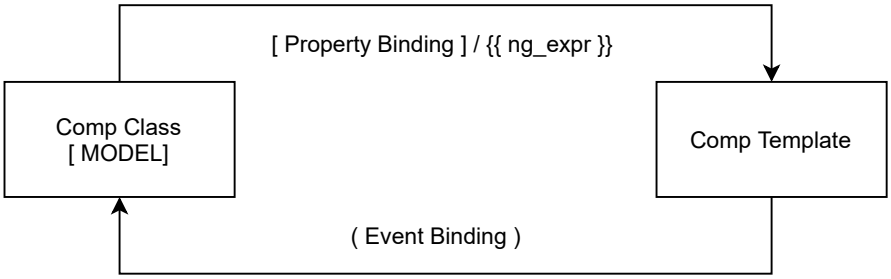


- Babel
- Trescuer
- Typescript Compiler

> npm install typescript -g

Angular Bootstrap Process



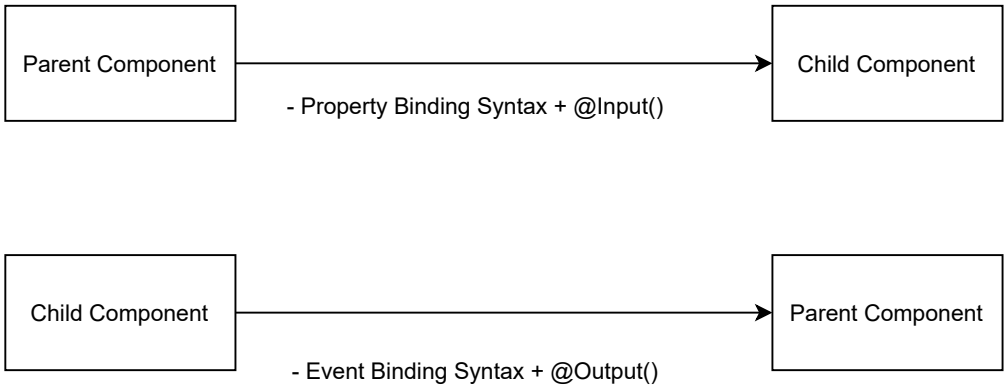


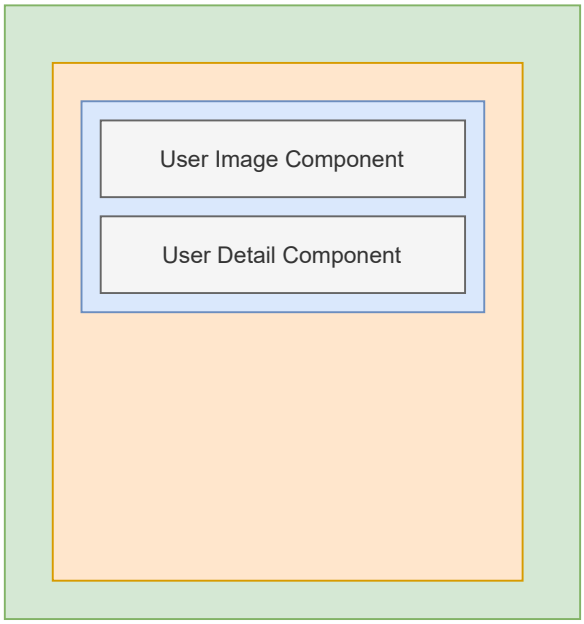
2 way data binding

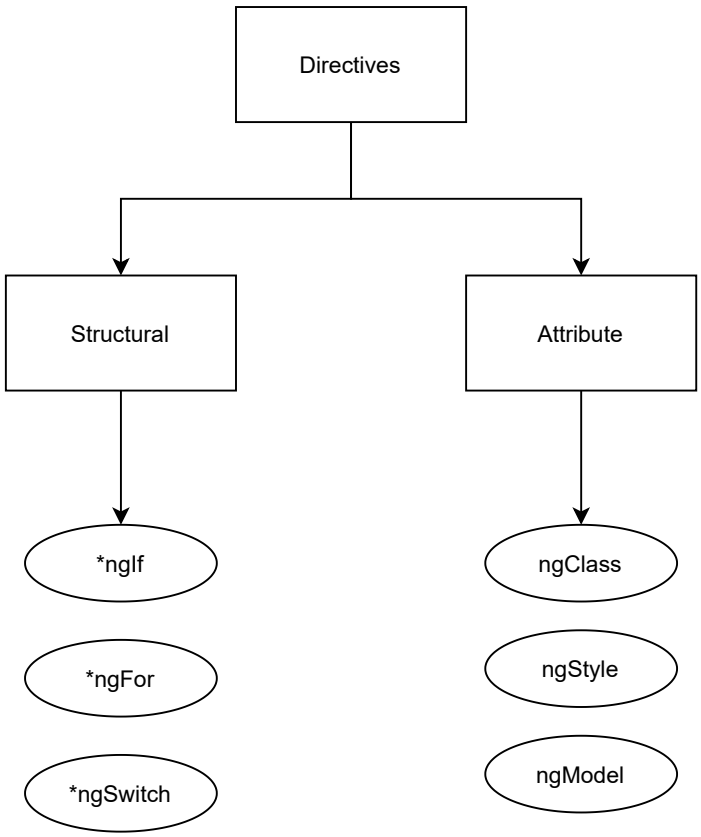
- Clubbing the [PB] + (EB)

- Banana in the Box - [(ngModel)]

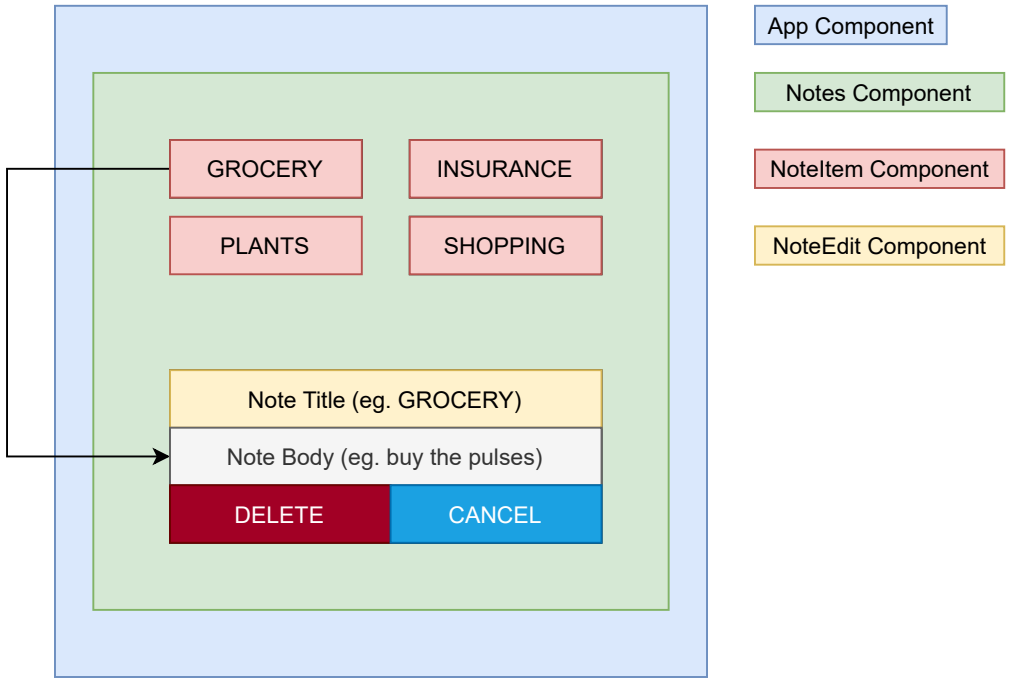
Component Communication






Directives
Structural
Attribute
*ngIf
*ngFor
*ngSwitch
ngClass
ngStyle
ngModel

Notes App



The Notes App - Instructions

- App should contain the notes collections as follows :

```
[
  { id : 1, title : "grocery", body : "to buy the pulses"},
  { id : 2, title : "shopping", body : "to buy the jeans"}
  { id : 3, title : "plants", body : "to pot the plants"},
  { id : 3, title : "insurance", body : "to renew car insurance"},
 ]
```

- Should create three components mentioned in picture :
    a. Notes : Container component, will contains NoteItem and NoteEdit Component
    b. NoteItem : displays individual item present in notes collection
    c. NoteEdit : display the note body and enables to delete the note from the collection

- Should display NoteEdit component on click of any of the NoteItem

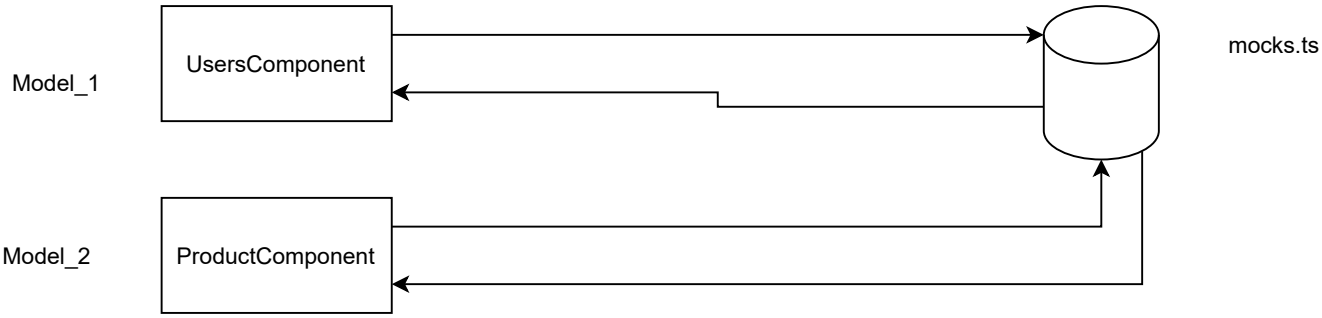
- Should use @Input / @Output for component communication.

Form & Form element - State and Classes

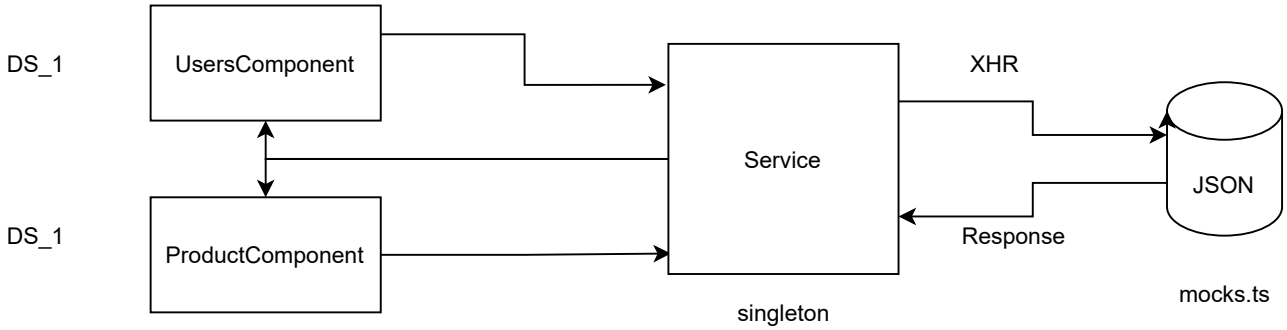
- ngValid | ngInvalid

- ngPristine | ngDirty

- ngTouched | ngUntouched



data service log service auth service

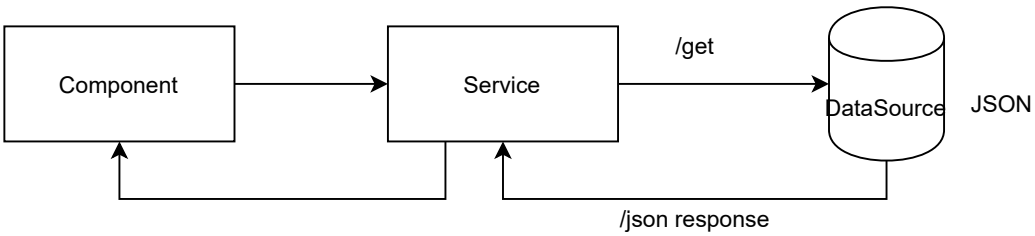


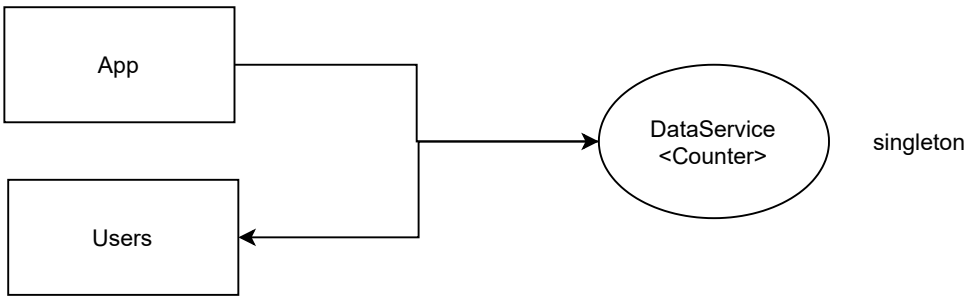
singleton

business login

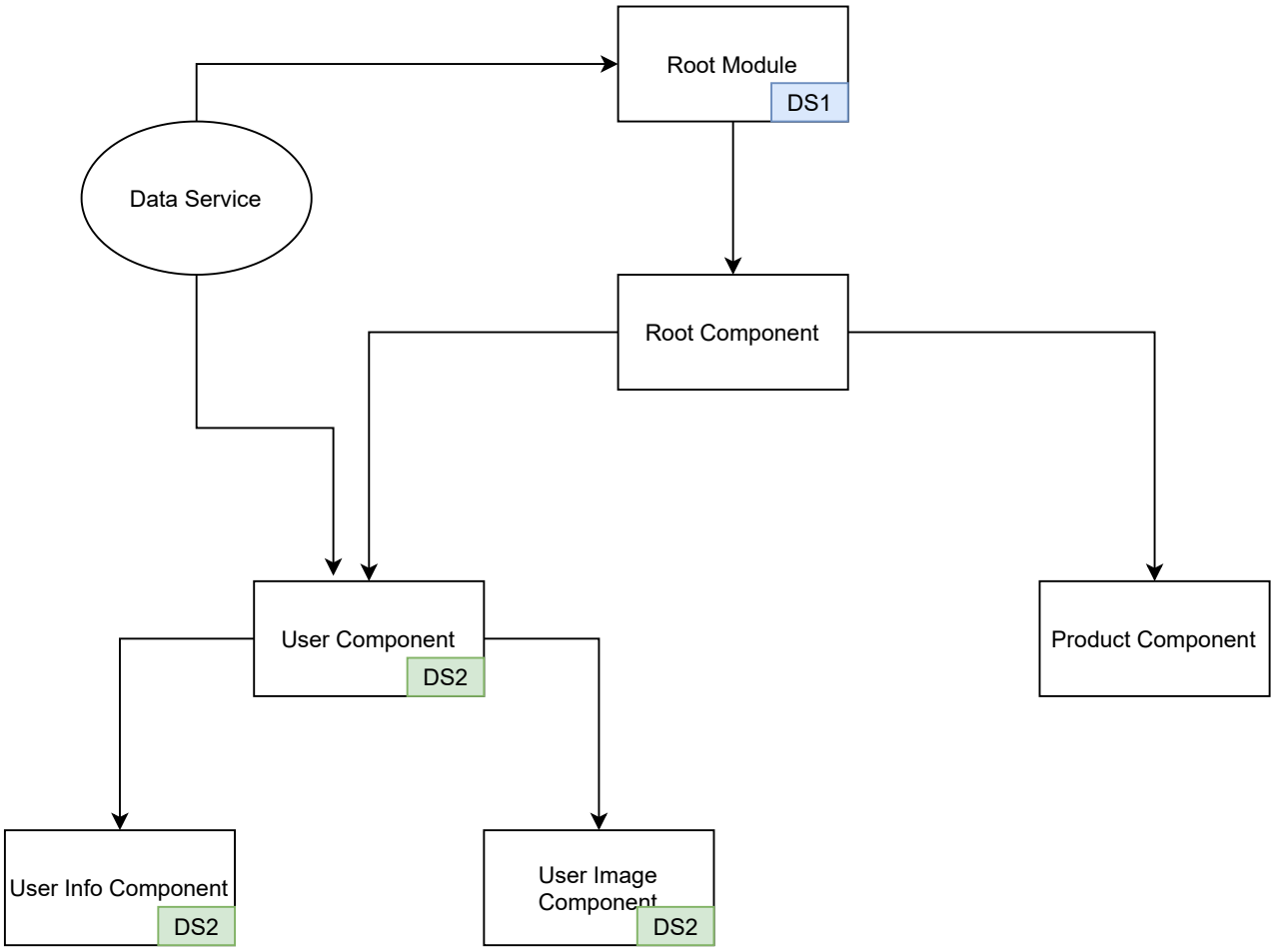
Service Principles

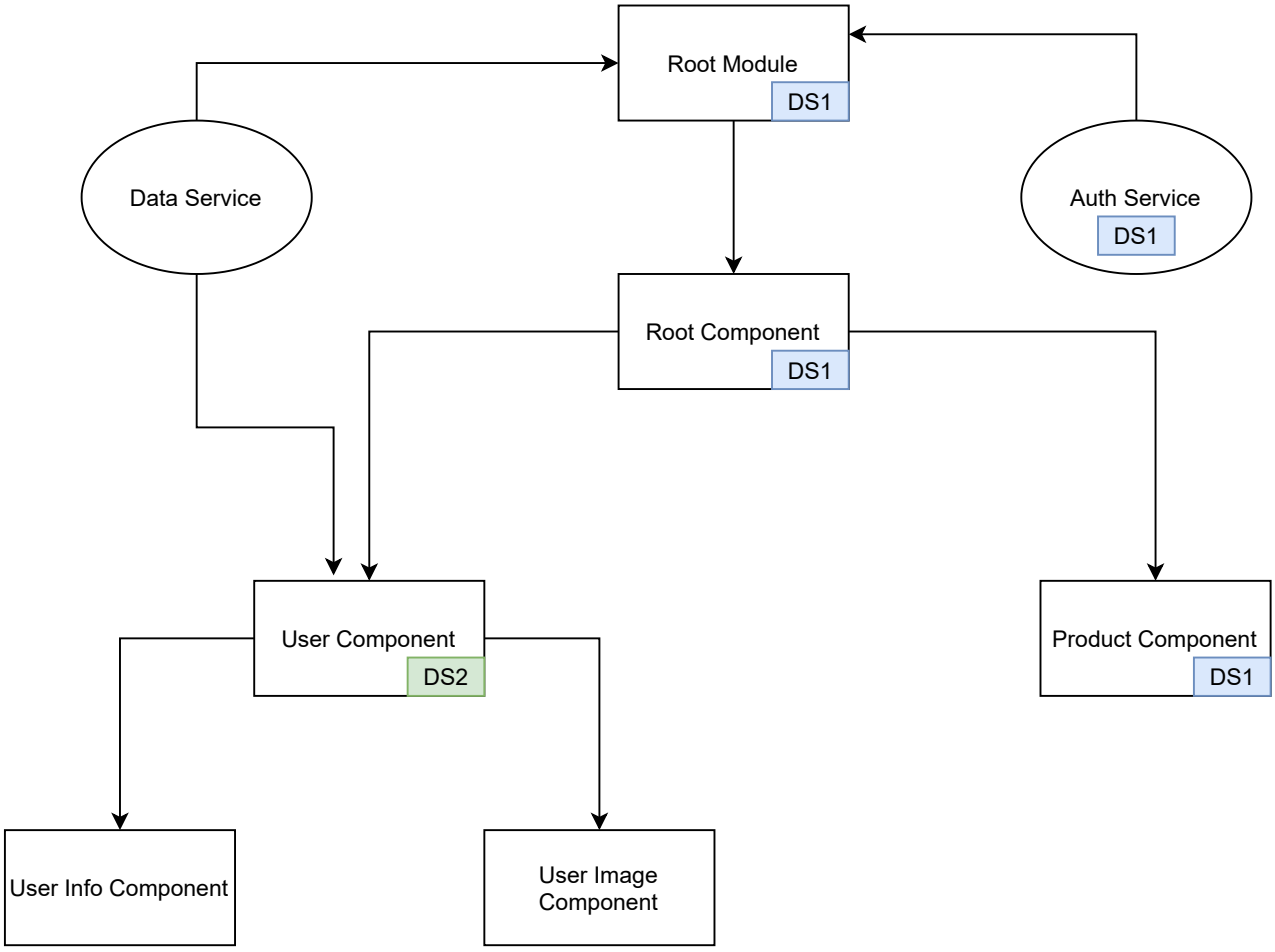
- Single Responsibility
- Dependency Injection

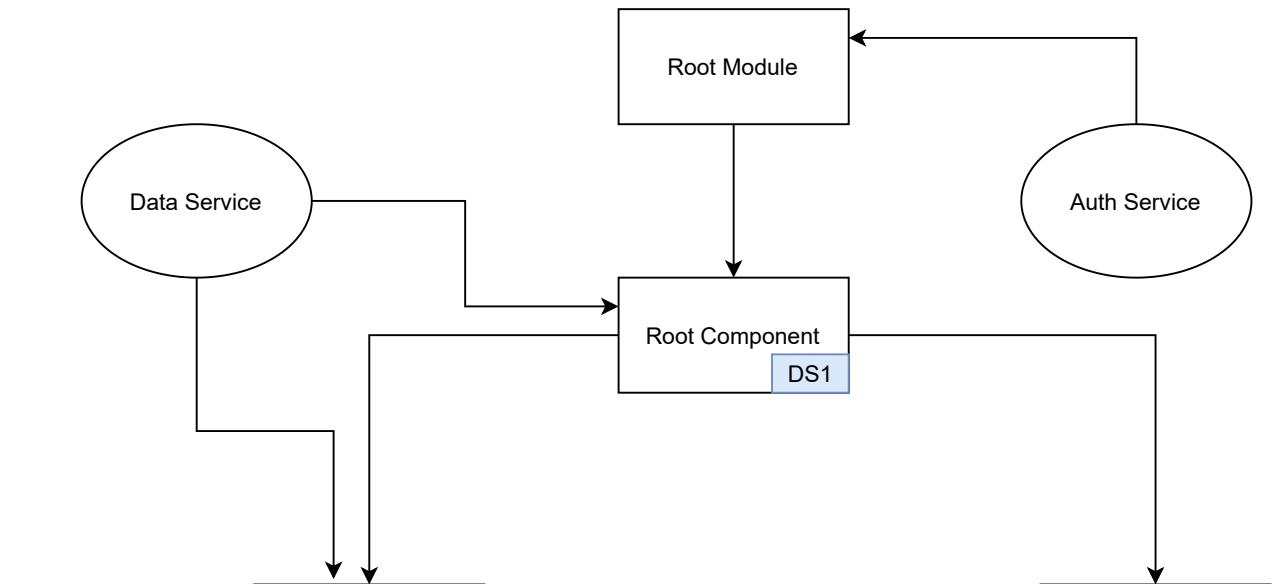


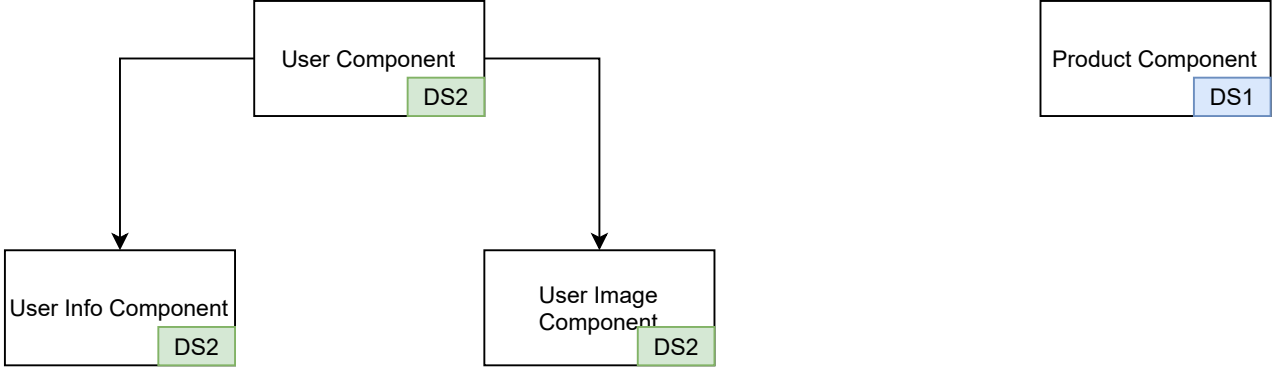


Multiple Service Registration




Root Module
DS1
Data Service
Auth Service
DS1
Root Component
DS1
User Component
DS2
Product Component
DS1
User Info Component
User Image
Component


Root Module
Data Service
Auth Service
Root Component
DS1


User Component
DS2
Product Component
DS1
User Info Component
DS2
User Image Component
DS2

Handling JS Async -

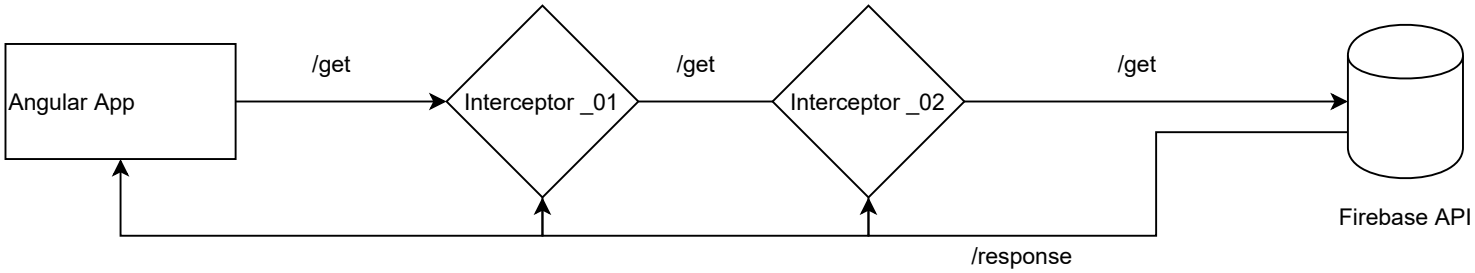
- Callback Functions

- Promises

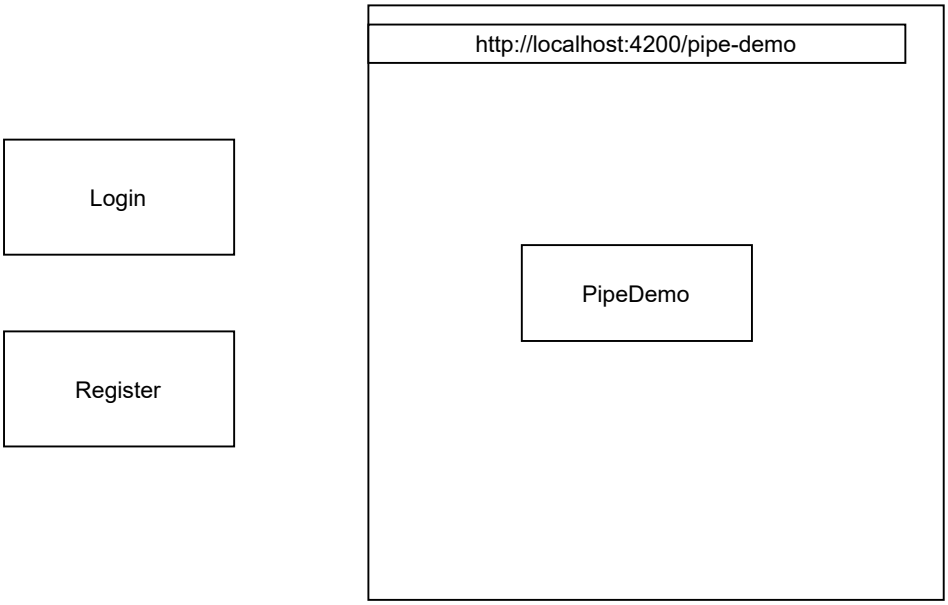
- Async ... await

- Observables

Interceptors



http://localhost:4200 - root route - /



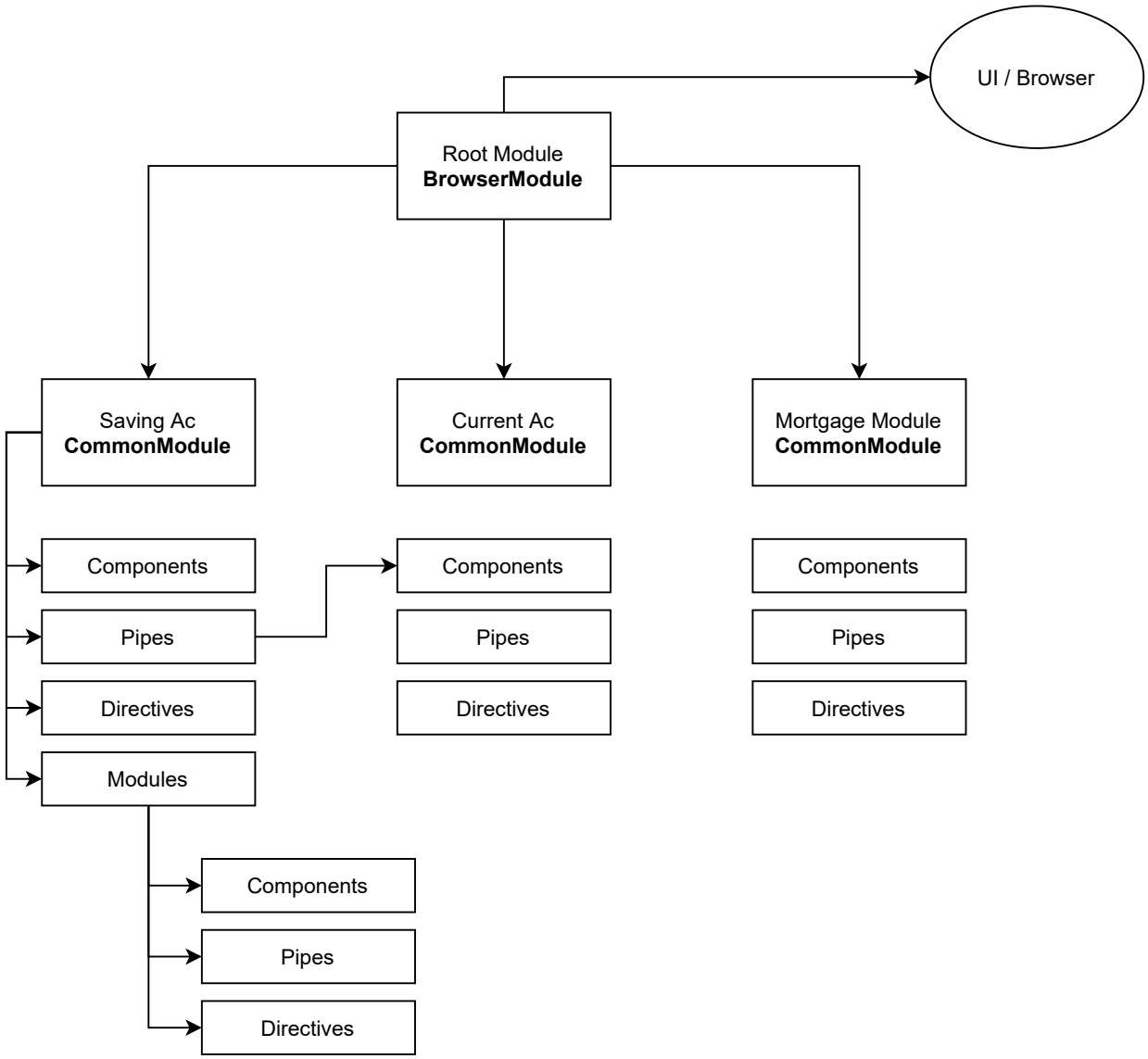
Routing Terminologies

- Routes : route configuration
- RouterModule : provides route config
- RouterLink : create links to change URL
- RouterOutlet : provide space on template
- RouteParameter : /
- QueryParameter : ?
- ActivatedRoute : info in URL

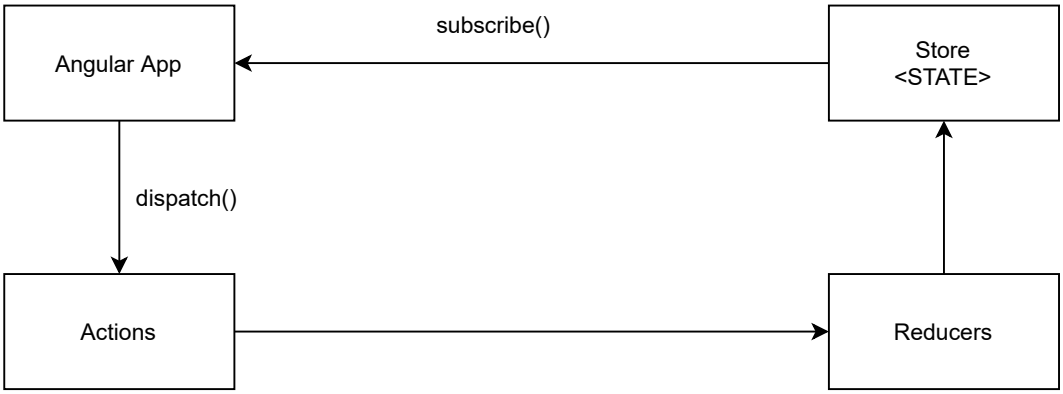
Hash (#) URL vs HTML5 URL / Regular URLs

http://localhost:4200#register ngv1.x
http://localhost:4200/login ng@latest

http://lcoalhost:4200/product/overview/99
http://lcoalhost:4200/product/spec?name=iPhone


UI / Browser
Root Module
BrowserModule
Saving Ac
CommonModule
Current Ac
CommonModule
Mortgage Module
CommonModule
Components
Pipes
Directives
Modules
Components
Pipes
Directives
Components
Pipes
Directives
Components
Pipes
Directives

Redux unidirectional flow of data



{ type : "", payload : "", data :"", value: "" }

(previousState, action) => newState