

Navigation and Routing Basics

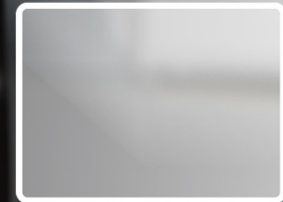
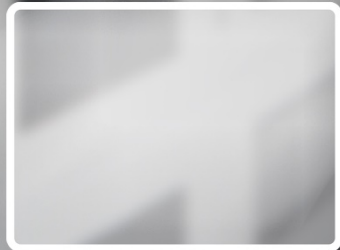
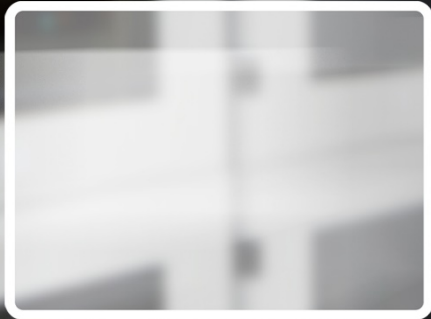


Deborah Kurata

CONSULTANT | SPEAKER | AUTHOR

@deborahkurata | blogs.msmvps.com/deborahk/





Module Overview



How Does Routing Work?

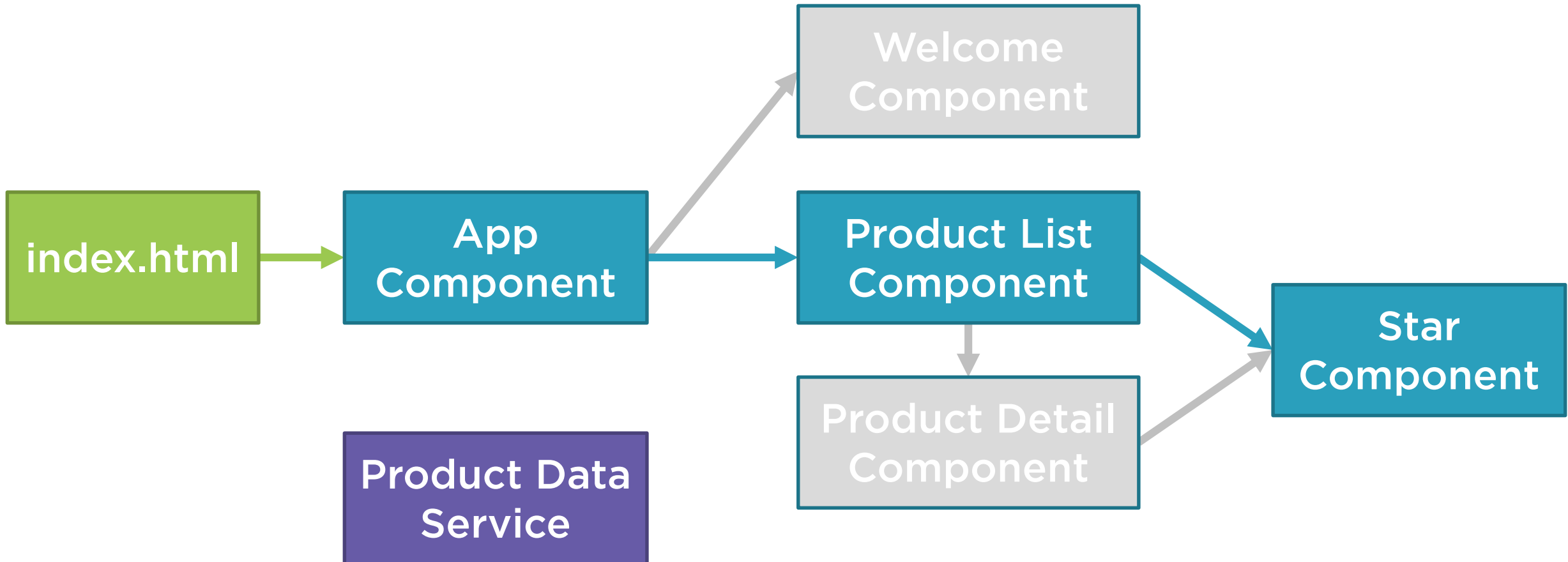
Configuring Routes

Tying Routes to Actions

Placing the Views



Application Architecture



How Routing Works

```
▼ <pm-app>
  ▼ <div>
    ▶ <nav class="navbar navbar-default">...</nav>
    ▼ <div class="container">
      ::before
      <router-outlet></router-outlet>
      ▼ <ng-component _ngghost-jfk-3>
        ▼ <div _ngcontent-jfk-3 class="panel panel-primary">
          <div _ngcontent-jfk-3 class="panel-heading">
            Product List
          </div>
          ▼ <div _ngcontent-jfk-3 class="panel-body">
            ::before
            ▶ <div _ngcontent-jfk-3 class="row">...</div>
            ▼ <div _ngcontent-jfk-3 class="table-responsive">
              ▼ <table _ngcontent-jfk-3 class="table">
                ▶ <thead _ngcontent-jfk-3>...</thead>
                ▶ <tbody _ngcontent-jfk-3>...</tbody>
              </table>
            </div>
            ::after
          </div>
        </div>
      </ng-component>
      ::after
    </div>
  </div>
</pm-app>
```

Configure a route for each component

Define options/actions

Tie a route to each option/action

Activate the route based on user action

Activating a route displays the component's view



How Routing Works

The screenshot shows a web browser at `localhost:3000` displaying the 'Acme Product Management' application. The navigation bar includes 'Acme Product Management', 'Home', and 'Product List'. A callout points to the 'Product List' link, showing the HTML: `Product List`. Another callout shows the routing configuration: `{ path: 'products', component: ProductListComponent }`. A third callout shows the component definition in `product-list.component.ts`:

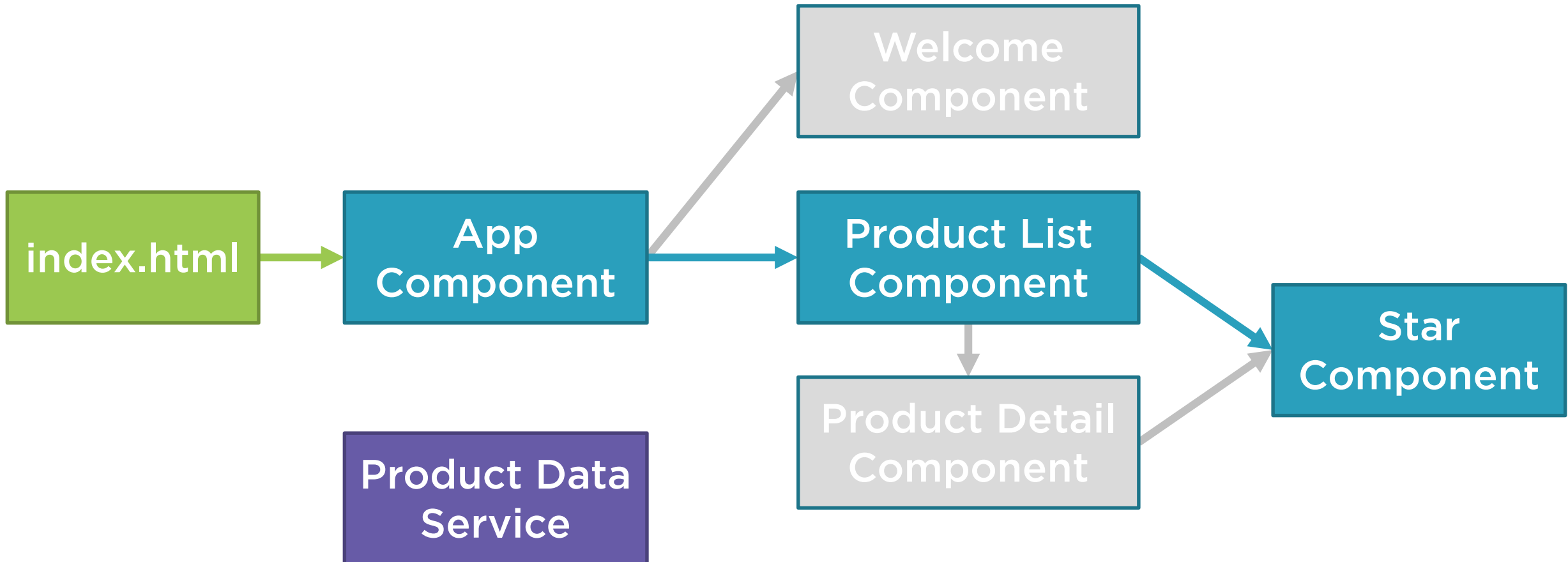
```
import { Component } from '@angular/core';

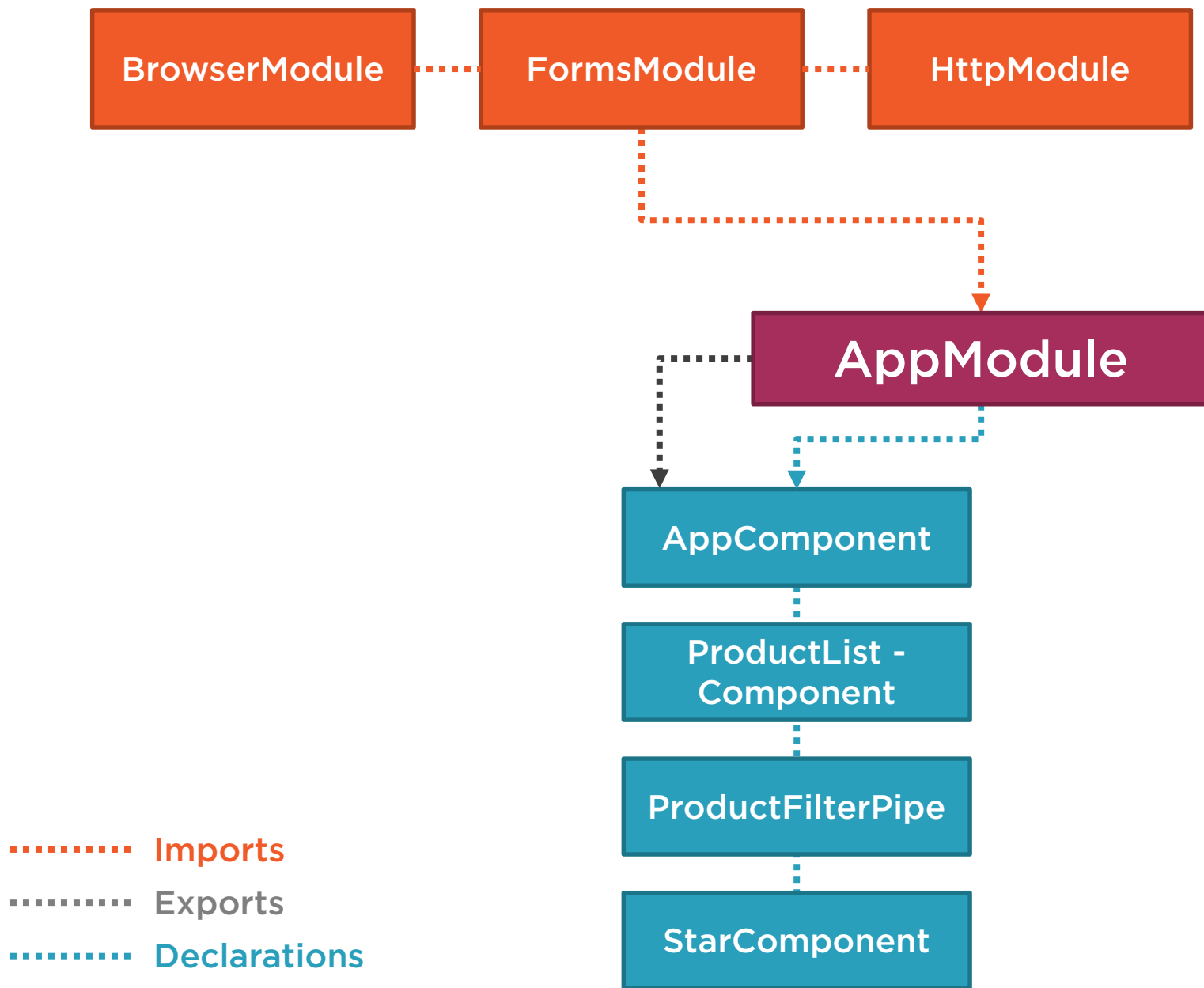
@Component({
  templateUrl: 'product-list.component.html'
})
export class ProductListComponent { }
```

Below the browser, a callout shows the router outlet: `<router-outlet></router-outlet>`.



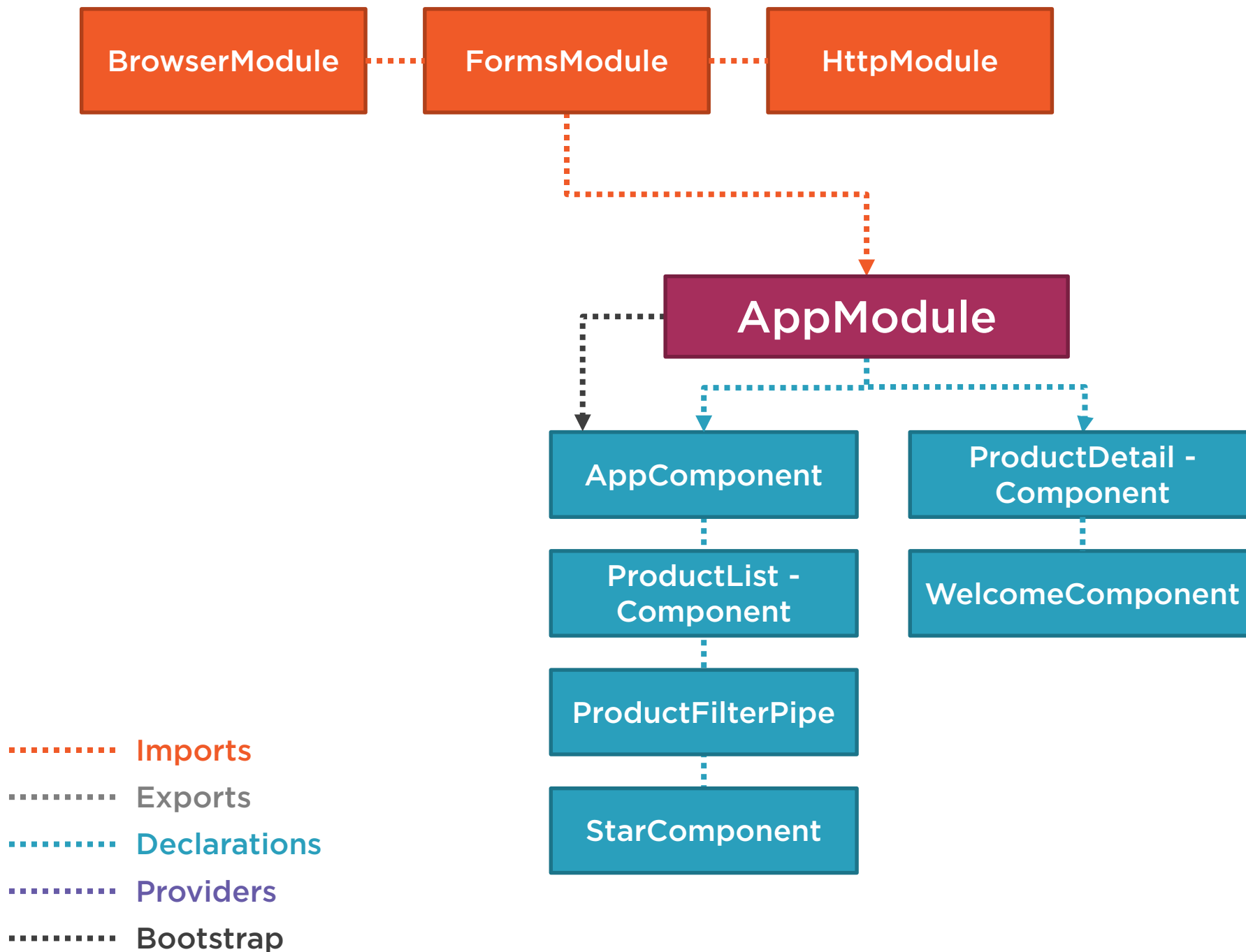
Application Architecture

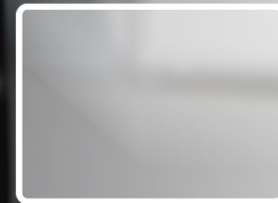
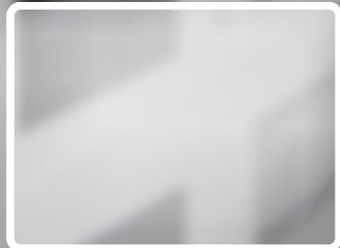
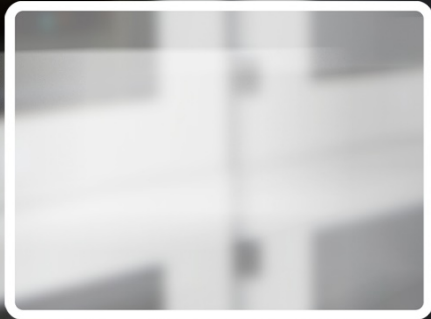




- Imports
- Exports
- Declarations
- Providers
- Bootstrap







'products', ProductListComponent

'product/:id', ProductDetailComponent

'welcome', WelcomeComponent



Configuring Routes

app.module.ts

```
...
import { RouterModule } from '@angular/router';

@NgModule({
  imports: [
    BrowserModule,
    FormsModule,
    HttpClientModule,
    RouterModule
  ],
  declarations: [
    ...
  ],
  bootstrap: [ AppComponent ]
})
export class AppModule { }
```



Configuring Routes

app.module.ts

```
...
import { RouterModule } from '@angular/router';

@NgModule({
  imports: [
    BrowserModule,
    FormsModule,
    HttpClientModule,
    RouterModule.forRoot([])
  ],
  declarations: [
    ...
  ],
  bootstrap: [ AppComponent ]
})
export class AppModule { }
```



Configuring Routes

app.module.ts

```
...
import { RouterModule } from '@angular/router';

@NgModule({
  imports: [
    BrowserModule,
    FormsModule,
    HttpClientModule,
    RouterModule.forRoot([], { useHash: true })
  ],
  declarations: [
    ...
  ],
  bootstrap: [ AppComponent ]
})
export class AppModule { }
```



Configuring Routes

```
[  
  { path: 'products', component: ProductListComponent },  
  { path: 'product/:id', component: ProductDetailComponent },  
  { path: 'welcome', component: WelcomeComponent },  
  { path: '', redirectTo: 'welcome', pathMatch: 'full' },  
  { path: '**', component: PageNotFoundComponent }  
]
```



Navigating the Application Routes



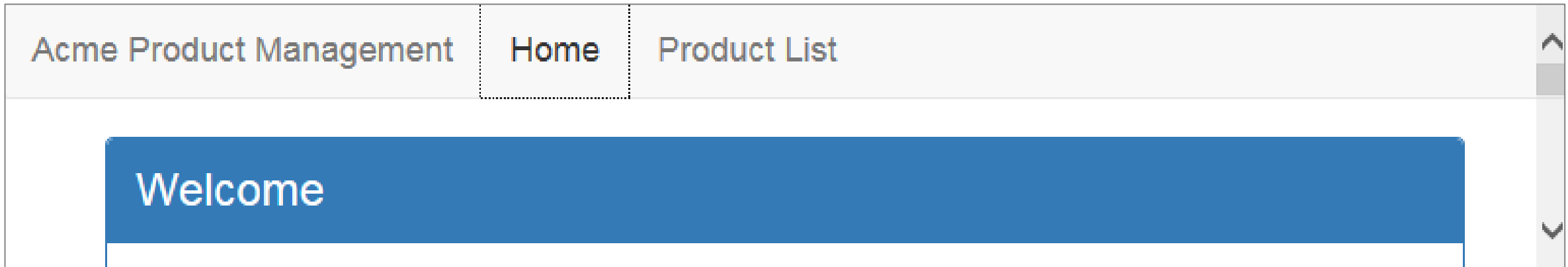
Menu option, link, image or button that activates a route

Typing the Url in the address bar / bookmark

The browser's forward or back buttons



Tying Routes to Actions



Tying Routes to Actions

app.component.ts

...

```
@Component({  
  selector: 'pm-app',  
  template: `  
    <ul class='nav navbar-nav'>  
      <li><a>Home</a></li>  
      <li><a>Product List</a></li>  
    </ul>  
  `,  
})
```



Tying Routes to Actions

app.component.ts

```
...

@Component({
  selector: 'pm-app',
  template: `
    <ul class='nav navbar-nav'>
      <li><a [routerLink]="[' /welcome']">Home</a></li>
      <li><a [routerLink]="[' /products']">Product List</a></li>
    </ul>
  `
})
```



Placing the Views

app.component.ts

```
...

@Component({
  selector: 'pm-app',
  template: `
    <ul class='nav navbar-nav'>
      <li><a [routerLink]="[' /welcome']">Home</a></li>
      <li><a [routerLink]="[' /products']">Product List</a></li>
    </ul>
    <router-outlet></router-outlet>
  `
})
```



How Routing Works



Acme Product Management

Home

Product List

```
<a [routerLink]='[' /products' ] '>Product List</a>
```

```
{ path: 'products', component: ProductListComponent }
```

```
<router-outlet></router-outlet>
```

product-list.component.ts

```
import { Component } from '@angular/core';

@Component({
  templateUrl: 'product-list.component.html'
})
export class ProductListComponent { }
```



Checklist: Displaying Components



Nest-able components

- Define a selector
- Nest in another component
- No route

Routed components

- No selector
- Configure routes
- Tie routes to actions

Checklist: Doing Routing







Configure routes

Tie routes to actions

Place the view



Routing Checklist: Configuring Routes



Define the base element

Add RouterModule

- Add each route (RouterModule.forRoot)
- Order matters

path: Url segment for the route

- No leading slash
- '' for default route
- '**' for wildcard route

component

- Not string name; not enclosed in quotes



Routing Checklist: Tying Routes to Actions



Add the RouterLink directive as an attribute

- Clickable element
- Enclose in square brackets

Bind to a link parameters array

- First element is the path
- All other elements are route parameters

Routing Checklist: Placing the View



Add the RouterOutlet directive

- Identifies where to display the routed component's view
- Specified in the host component's template

Summary



How Does Routing Work?

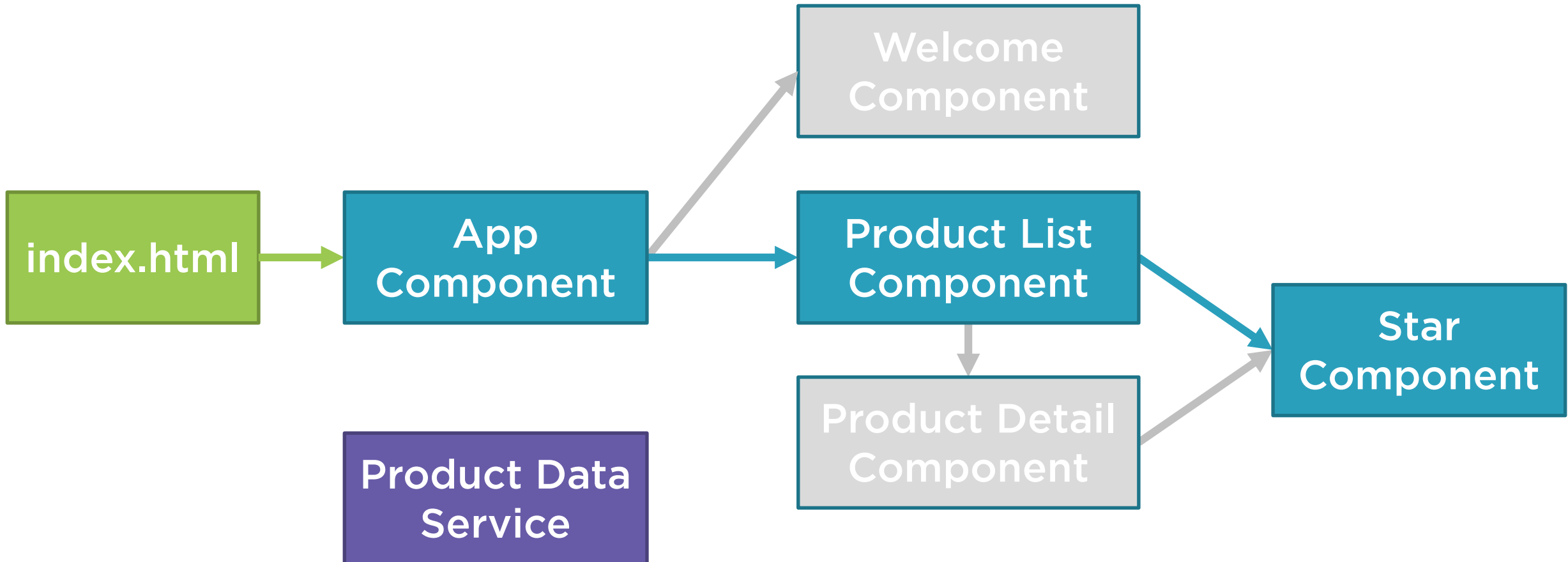
Configuring Routes

Tying Routes to Actions

Placing the Views



Application Architecture



Application Architecture

