

Routing to Features



Deborah Kurata

CONSULTANT | SPEAKER | AUTHOR | MVP | GDE

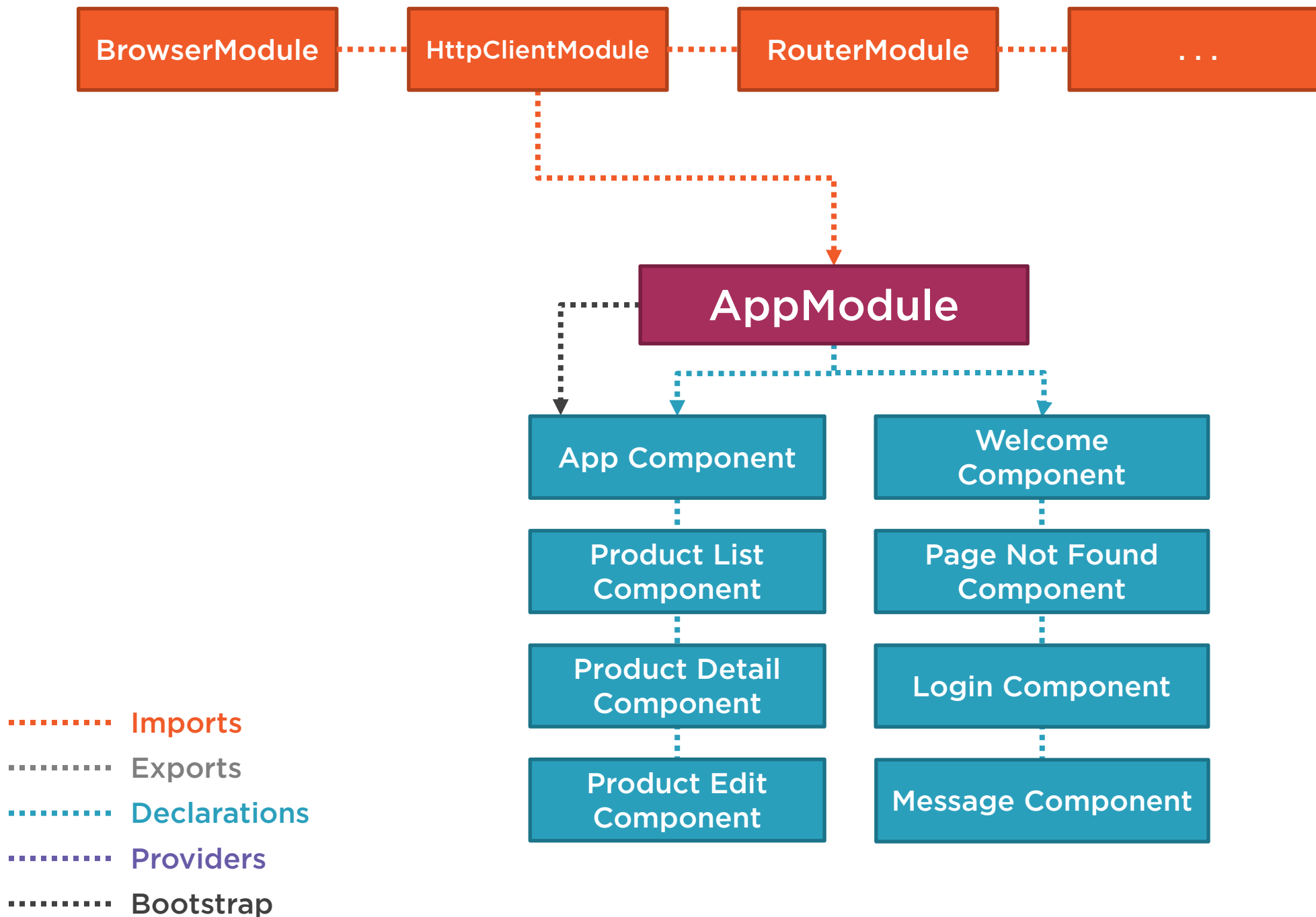
@deborahkurata | blogs.msmvps.com/deborahk/



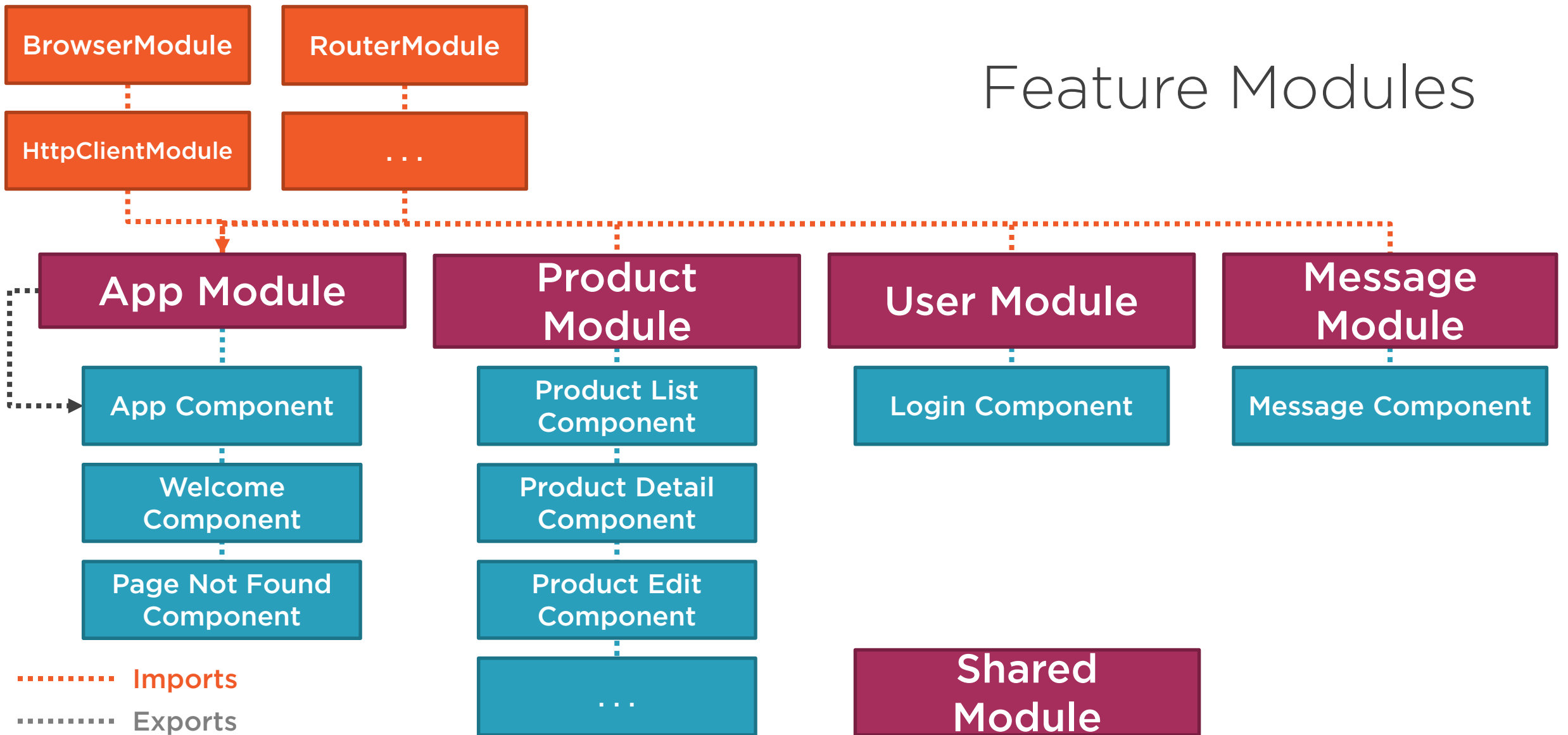
Feature Module

An Angular module created with the express purpose of organizing the components for a specific application feature area.





Feature Modules



..... Imports

..... Exports

..... Declarations

..... Providers

..... Bootstrap



Module Overview



Setting up for Feature Routing

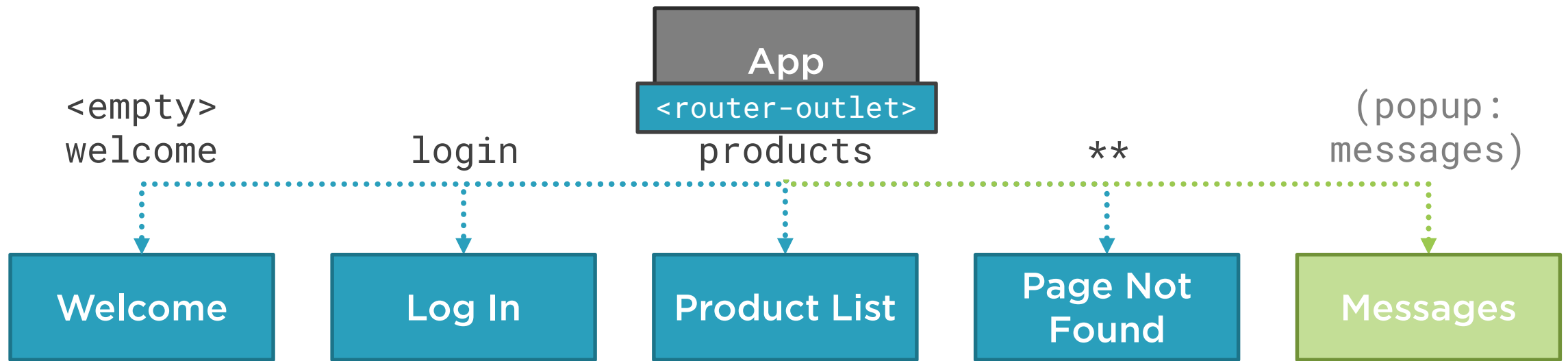
Route Path Naming Strategies

Activating a Route with Code

Accessing Feature Routes

Defining a Routing Module





- Primary Routes
- Secondary Routes
- Child Routes



Setting up for Feature Routing



Import Router

Configure routes

Activate routes



Importing the Angular Router

RouterModule

Service

Configuration

Directives



Importing the Angular Router

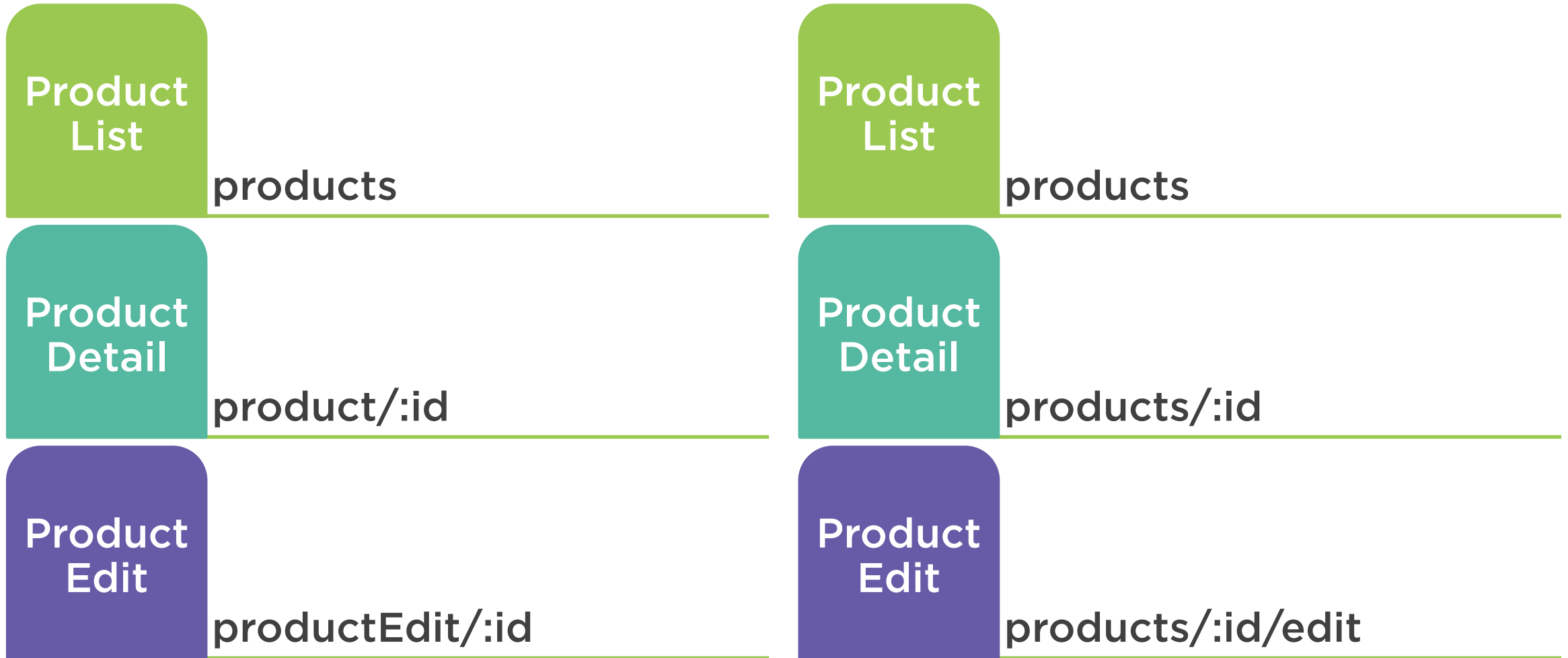
product.module.ts

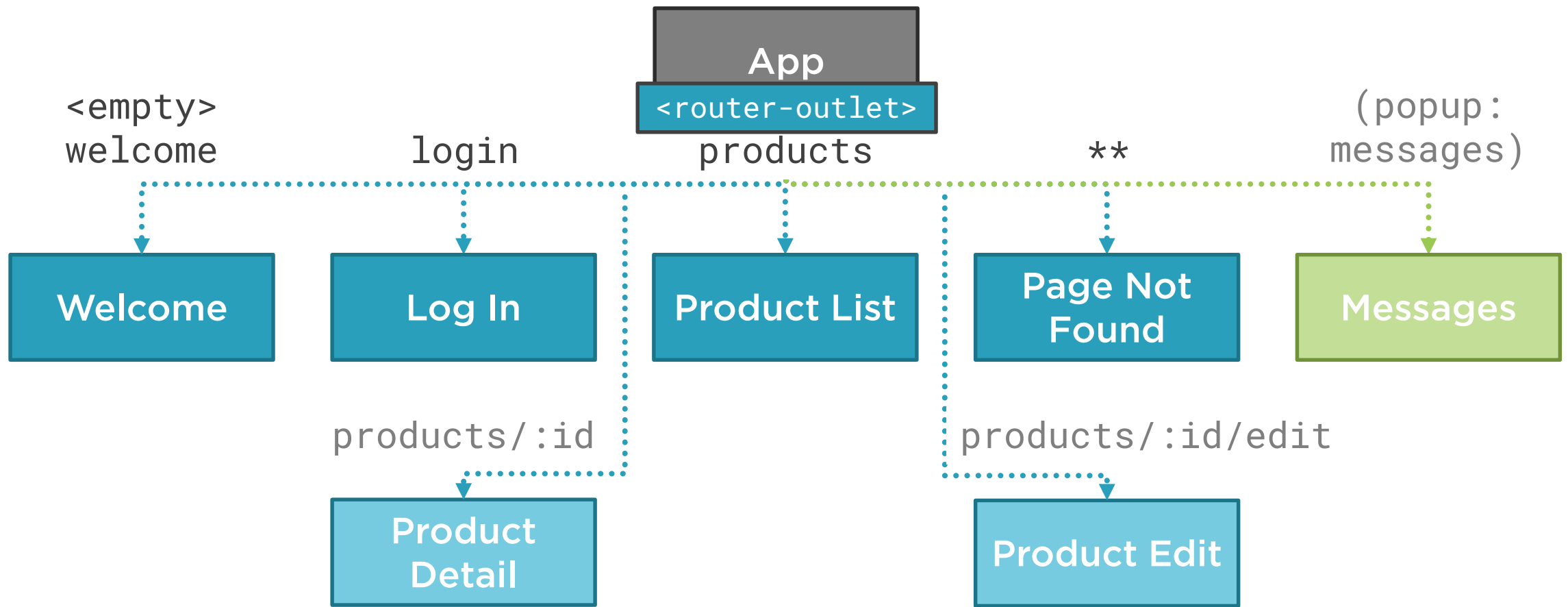
```
import { RouterModule } from '@angular/router';
...

@NgModule({
  imports: [
    SharedModule,
    RouterModule.forChild([])
  ],
  declarations: [
    ProductListComponent,
    ...
  ],
  ...
})
export class ProductModule { }
```



Route Path Naming Strategies

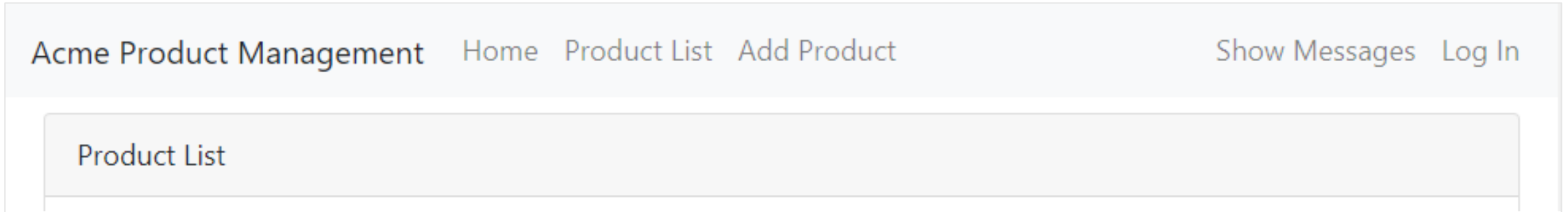




- Primary Routes
- Secondary Routes
- Child Routes



Activating a Route with Code



Activating a Route with Code

app.component.ts

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

@Component({...})
export class AppComponent {
  ...
  constructor(private router: Router) { }

  logOut(): void {
    // Do some processing
    this.router.navigate([' /welcome' ]);
  }
}
```



Activating a Route with Code

```
this.router.navigate([' /welcome' ]);           // Standard syntax
```

```
this.router.navigate(' /welcome' );           // Short-cut syntax
```

```
this.router.navigateByUrl(' /welcome' );       // Complete Url path
```

```
http://localhost:4200/products(popup:messages) // current route
```

```
http://localhost:4200/welcome(popup:messages) // navigate()
```

```
http://localhost:4200/welcome                  // navigateByUrl()
```



Demo



Activating a Route with Code



Demo



Accessing Feature Routes



Matching Routes

app.module.ts

```
imports: [  
  BrowserModule,  
  HttpClientModule,  
  RouterModule.forRoot([  
    { path: 'welcome',  
      component: WelcomeComponent },  
    { path: '', redirectTo: 'welcome',  
      pathMatch: 'full' },  
    { path: '**',  
      component: PageNotFoundComponent }  
  ]),  
  ProductModule,  
  UserModule,  
  MessageModule  
]
```

Route Paths

- **welcome**
- **"**
- ******
- **products**
- **login**



Matching Routes

app.module.ts

```
imports: [  
  BrowserModule,  
  HttpClientModule,  
  RouterModule.forRoot([  
    { path: 'welcome',  
      component: WelcomeComponent },  
    { path: '', redirectTo: 'welcome',  
      pathMatch: 'full' },  
    { path: '**',  
      component: PageNotFoundComponent }  
  ]),  
  ProductModule,  
  UserModule,  
  MessageModule  
]
```

Route Paths

- **products**
- **login**
- **welcome**
- **"**
- ******



Why Define a Routing Module?



- **Better organization**
- **Easier to find**
- **Separation of concerns**

Matching Routes

app.module.ts

```
imports: [  
  BrowserModule,  
  HttpClientModule,  
  AppRoutingModule,  
  ProductModule,  
  UserModule,  
  MessageModule  
]
```

Route Paths

- **welcome**
- **"**
- ******
- **products**
- **login**



Matching Routes

app.module.ts

```
imports: [  
  BrowserModule,  
  HttpClientModule,  
  ProductModule,  
  UserModule,  
  MessageModule,  
  AppRoutingModule  
]
```

Route Paths

- **products**
- **login**
- **welcome**
- **"**
- ******



Defining a Configuration Constant

app-routing.module.ts

```
const ROUTES = [
  { path: 'welcome', component: WelcomeComponent },
  { path: '', redirectTo: 'welcome', pathMatch: 'full' },
  { path: '**', component: PageNotFoundComponent }
];

@NgModule({
  imports: [
    RouterModule.forRoot(ROUTES)
  ],
  exports: [ RouterModule ]
})
export class AppRoutingModule { }
```



Routing to Features Checklist: Configuration







Import RouterModule

- Be sure to use RouterModule.forChild()

Configure the routes

Order matters!

```
@NgModule({  
  imports: [  
    ...,  
    RouterModule.forChild([ { path: 'products',  
                           component: ProductListComponent } ])  
  ], ...  
})
```

Routing to Features Checklist: Naming Routes



Use a common root path name for related feature routes

- products
- products/:id
- products/:id/edit

Routing to Features Checklist: Activate with Code



Import the Router

Add a dependency on the Router service

- As a constructor parameter

Use the Router service navigate method

Pass in a link parameters array

- First element is the root Url segment
- All other elements are route parameters or additional Url segments

```
this.router.navigate([' /welcome' ]);
```

```
this.router.navigateByUrl(' /welcome' );
```



Routing to Features Checklist: Routing Modules



Separate out routes to their own routing module

Keep route path order in mind



Summary



Setting up for Feature Routing

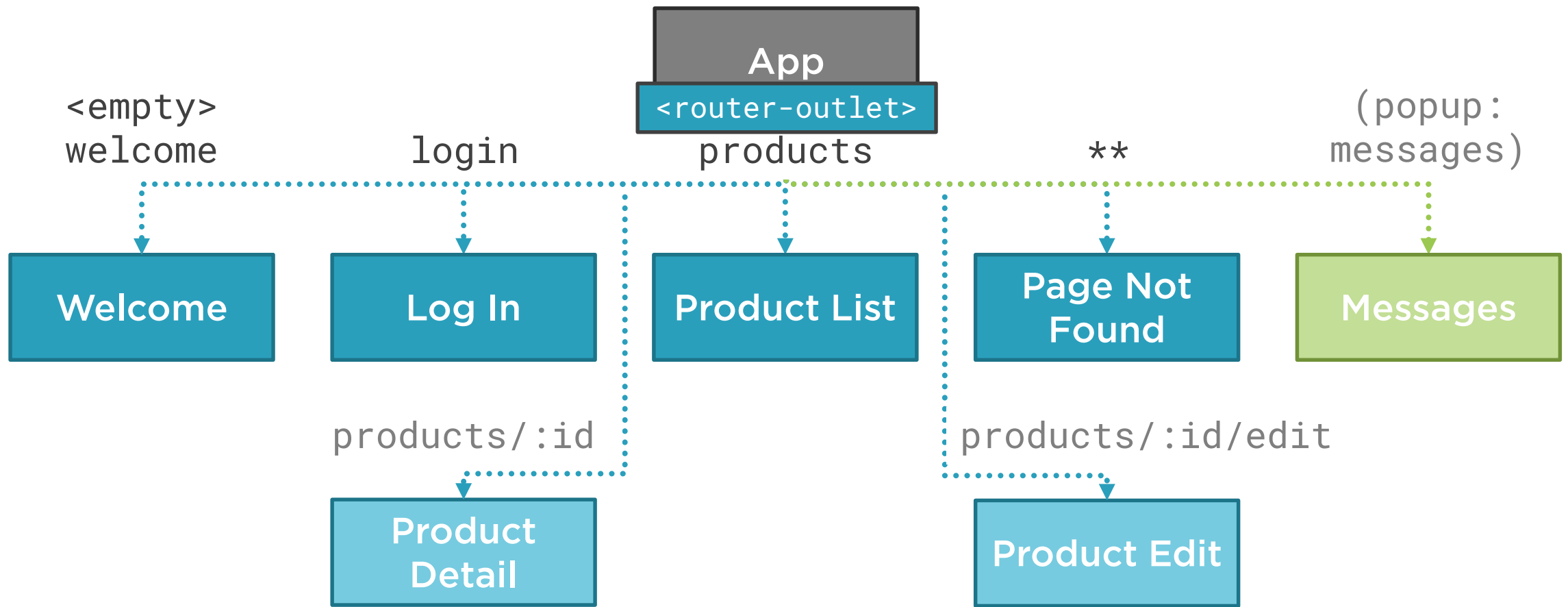
Route Path Naming Strategies

Activating a Route with Code

Accessing Feature Routes

Defining a Routing Module





- Primary Routes
- Secondary Routes
- Child Routes

