



ANGULAR  
**ARCHITECTS**  
INSIDE KNOWLEDGE

# Routing & Navigation

Alex Thalhammer

# Outline

- Motivation
- Configuration
- Routing parameters

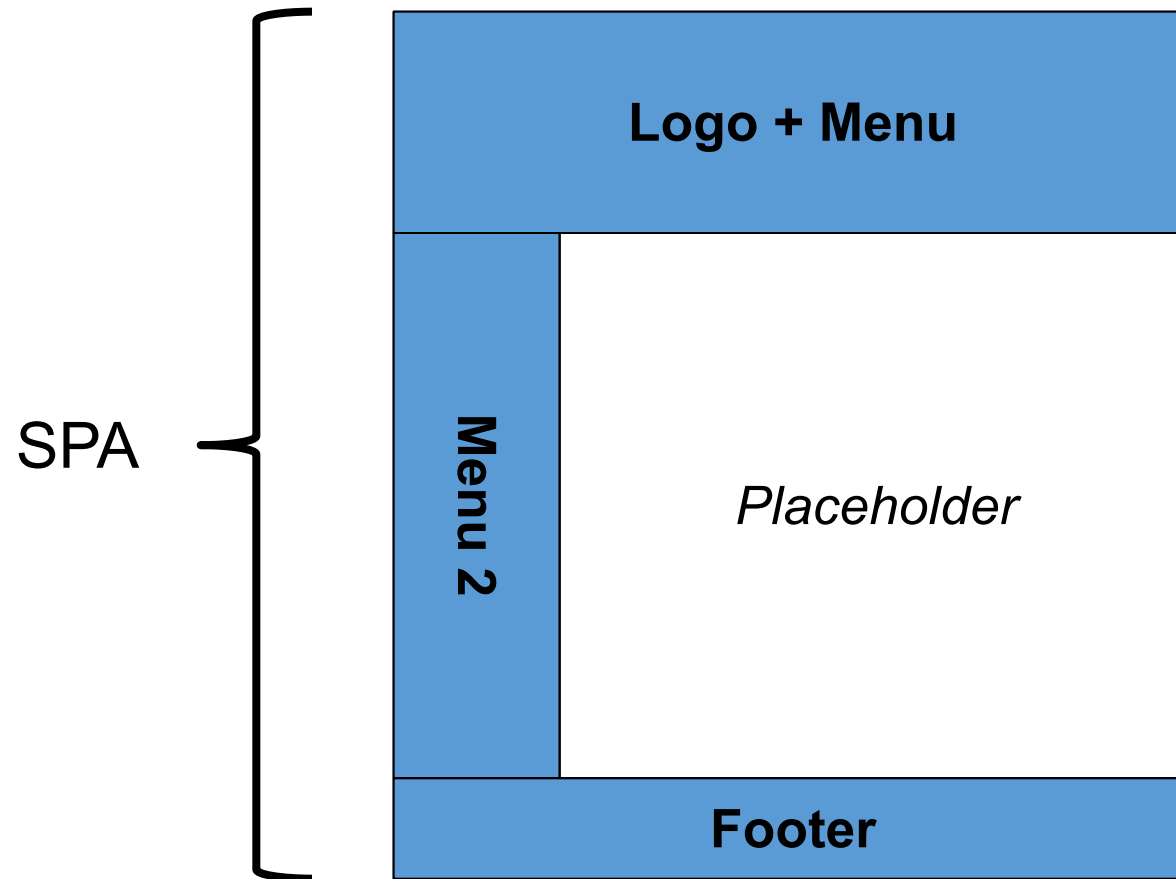


# Motivation

- SPAs → single page application
- Simulate pages → routes
- URL should direct to the routed component
  - Menus & Bookmarks
  - Sharing (social)
  - **Back button!**

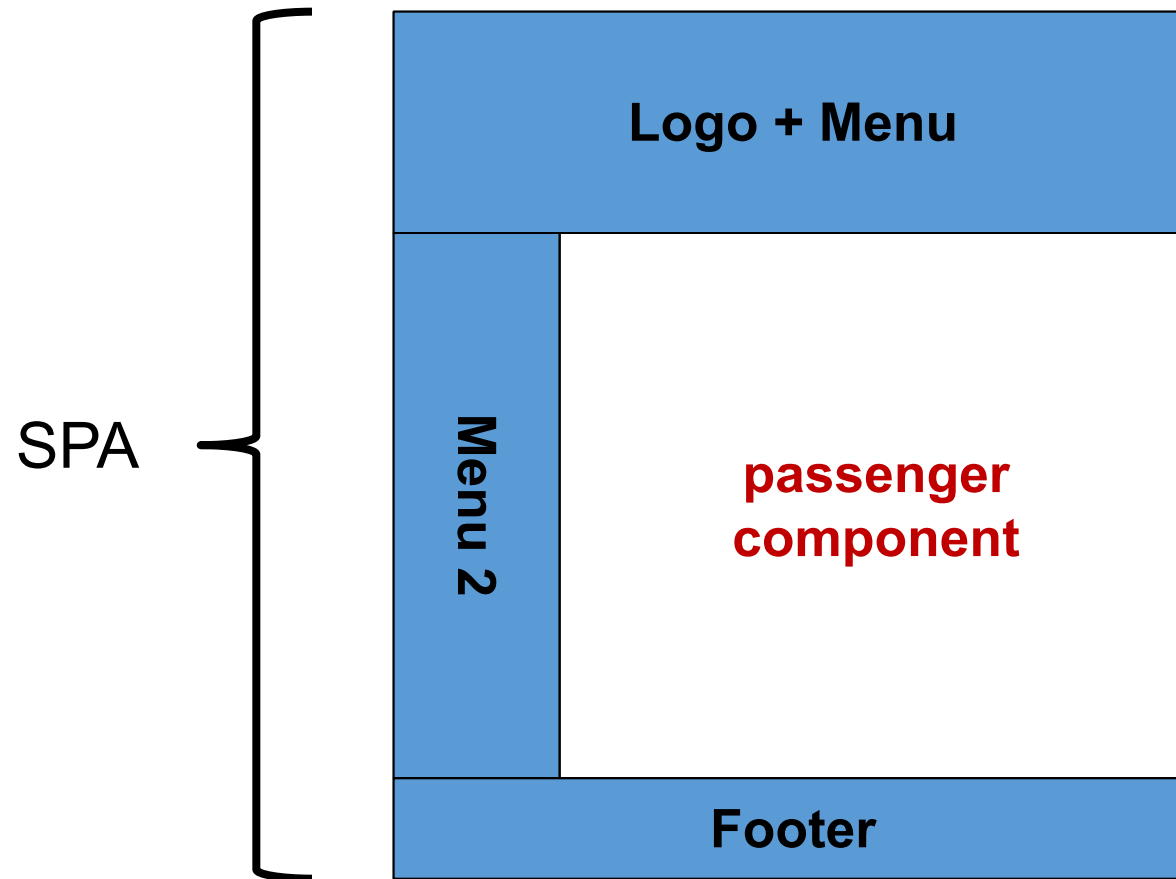


# Routing in Angular



# Routing in Angular

/flight-demo/**passenger**



# Routing via hash fragment

- /flight-demo#passenger
- Hash-Fragment is never sent to the server

# Routing via History API

- /flight-demo/**passenger**
- Whole URL will be send to the server
- Server responds with the SPA
- Server can optionally render the initial view
  - Performance, SEO, ...
- SPA informs the browser which part of the URL belongs:
  - to the server
  - to the Angular Route







# Angular Router



# Preparation

- Angular Module: RouterModule (@angular/router)
- Definition of the limit between URL and Angular Route with the History API:
  - Options to define:
    - angular.json
    - <base href="/"> in index.html
    - Token: APP\_BASE\_HREF (@angular/common)

# Configuration

```
export const APP_ROUTES: Routes = [  
  {  
    path: 'home',  
    component: HomeComponent  
  },  
  {  
    path: 'flight-search',  
    component: FlightSearchComponent  
  }  
]
```



# Configuration

```
export const APP_ROUTES: Routes = [  
  {  
    path: '',  
    redirectTo: 'home',  
    pathMatch: 'full'  
  },  
  {  
    path: 'home',  
    component: HomeComponent  
  },  
  {  
    path: 'flight-search',  
    component: FlightSearchComponent  
  }  
]
```



# Configuration

```
export const APP_ROUTES: Routes = [  
  {  
    path: '',  
    redirectTo: 'home',  
    pathMatch: 'full'  
  },  
  {  
    path: 'home',  
    component: HomeComponent  
  },  
  [...]  
  {  
    path: '**',  
    redirectTo: 'home'  
  }  
]
```

←----- Last route!!!



# Configuration

```
// app.module.ts
@NgModule({
  imports: [
    BrowserModule,
    HttpClientModule,
    FormsModule,
    RouterModule.forRoot(APP_ROUTES)
  ],
  [...],
})
export class AppModule {}
```

**for feature module: forChild**

**for root module**



# View von AppComponent

```
<a [routerLink]=" '/' ">Home</a>  
<a [routerLink]=" '/flight-search' ">Search flight</a>  
  
<div>  
  <router-outlet></router-outlet>  
</div>
```





# View von AppComponent

```
<a routerLink="/">Home</a>  
<a routerLink="/flight-search">Search flight</a>  
  
<div>  
  <router-outlet></router-outlet>  
</div>
```



# Programmatic routing

```
export class AppComponent {  
  
    constructor(private router: Router) {}  
  
    navigateToFlightSearch(): void {  
        this.router.navigate(['/flight-search']);  
    }  
  
}
```



# DEMO





# Routing with parameters

# Parameters

- `passenger/7`
- `passenger/7;details=true;page=7`
- `passenger/7;details=true;page=7/flights`



# Configuration

```
const APP_ROUTES: Routes = [  
  [...]  
  {  
    path: 'flight-search',  
    component: FlightSearchComponent  
  },  
  {  
    path: 'flight-edit/:id',  
    component: FlightEditComponent  
  }  
]
```





# Parse parameters

```
export class FlightEditComponent {  
  public id: string;  
  
  constructor(private route: ActivatedRoute) {  
  
    ngOnInit(): void {  
      this.route.paramMap.subscribe(  
        params => {  
          this.id = params.get('id');  
          [...]  
        }  
      );  
    }  
  }  
  [...]  
}
```



# Links of routes with parameters

```
<a [routerLink]="['/flight-edit', flight.id]'>Edit</a>
```

# Links of routes with parameters

```
<a [routerLink]="['/flight-edit', flight.id, {showDetails:true}]">Edit</a>
```



# DEMO



# LAB

