Recapitulando...

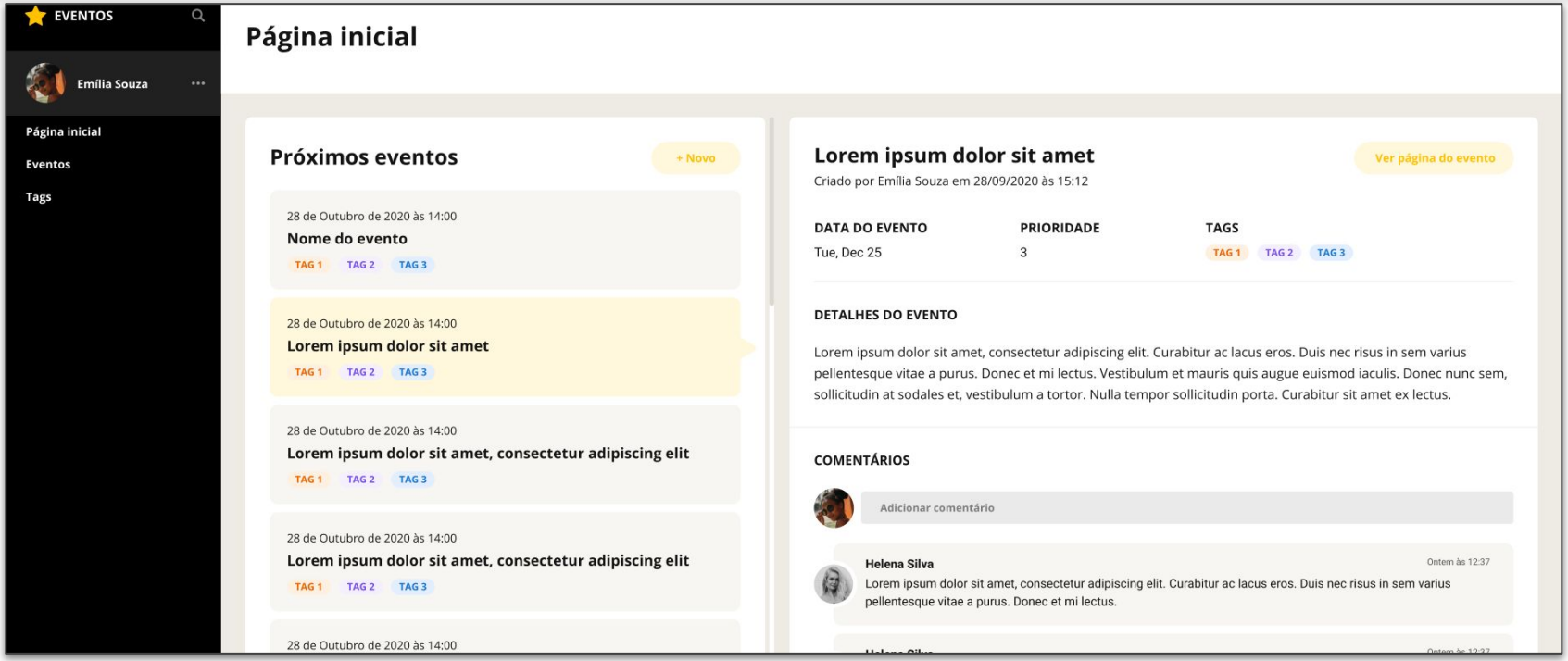
Serviços

Pipes

- Pipes são funções simples que podem ser usadas em expressões de templates (interpolação);
- Podem transformar strings, datas, valores monetários e vários outros tipos de dados;
- Aceitam um valor de entrada e retornam um valor transformado;
- São úteis porque evitam repetição de código, um pipe pode ser utilizado na aplicação inteira

```
<p>The hero's birthday is {{ birthday | date }}</p>
```

Projeto


EVENTOS
Emília Souza
Página inicial
Eventos
Tags
Página inicial
Próximos eventos
+ Novo
28 de Outubro de 2020 às 14:00
Nome do evento
TAG 1 TAG 2 TAG 3
28 de Outubro de 2020 às 14:00
Lorem ipsum dolor sit amet
TAG 1 TAG 2 TAG 3
28 de Outubro de 2020 às 14:00
Lorem ipsum dolor sit amet, consectetur adipiscing elit
TAG 1 TAG 2 TAG 3
28 de Outubro de 2020 às 14:00
Lorem ipsum dolor sit amet, consectetur adipiscing elit
TAG 1 TAG 2 TAG 3
28 de Outubro de 2020 às 14:00
Lorem ipsum dolor sit amet
Criado por Emília Souza em 28/09/2020 às 15:12
Ver página do evento
DATA DO EVENTO
Tue, Dec 25
PRIORIDADE
3
TAGS
TAG 1 TAG 2 TAG 3
DETALHES DO EVENTO
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur ac lacus eros. Duis nec risus in sem varius pellentesque vitae a purus. Donec et mi lectus. Vestibulum et mauris quis augue euismod iaculis. Donec nunc sem, sollicitudin at sodales et, vestibulum a tortor. Nulla tempor sollicitudin porta. Curabitur sit amet ex lectus.
COMENTÁRIOS
Adicionar comentário
Helena Silva
Ontem às 12:37
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur ac lacus eros. Duis nec risus in sem varius pellentesque vitae a purus. Donec et mi lectus.

Rotas

Single Page Application

- Aplicações que rodam em uma única página
- SPA: https://angular.io/
- Não SPA: https://medium.com/

Roteamento

https://meuprojeto.com/**eventos**

EventListComponent

Roteamento

https://meuprojeto.com/**eventos/5/edit**

Angular lê a rota

Angular identifica a rota

EventEditComponent

Roteamento

/eventos/5/edit

/eventos	EventListComponent
/events/:id	EventDetailComponent
/events/:id/edit	EventFormComponent

Criando rotas simples

- Pode ser criada tanto dentro do módulo quanto em um arquivo separado
  1. Criar arquivo com módulo de rotas da aplicação para ser exportado
     a. Rota raíz (forRoot) vs Rota de funcionalidade (forChild)
  2. Importar classe de rotas no módulo da aplicação
  3. Configurar tag <router-outlet></router-outlet
     a. Permite o Angular ter o suporte a rotas
     b. O componente da rota é renderizado em baixo da tag <router-outlet></router-outlet>

RouterLink

- Maneira de “navegar” entre as páginas da aplicação através de links

```
<div class="navbar">
 <a routerLink="/">Home</a>
 <a routerLink="/courses">Cursos</a>
 <a routerLink="/login">Login</a>
</div>
```

RouterLinkActive

- Maneira de saber qual rota/link está ativo

```
<div class="navbar">
 <a routerLinkActive="active"  routerLink="/home">Home</a>
 <a routerLinkActive="active"
routerLink="/courses">Cursos</a>
 <a routerLinkActive="active"  routerLink="/login">Login</a>
</div>
```

Lendo parâmetros de rotas

- Utilizamos a classe ActivatedRoute

Destino:

```
constructor(private route: ActivatedRoute) {
   this.courseId = route.snapshot.params['id'];
}
```

Origem:

```
<a routerLinkActive="active"
  [routerLink]="['course', courseId]">
     Detalhes  Curso
</a>
```

Lendo parâmetros de rotas

- Utilizamos a classe ActivatedRoute

Destino:

```
constructor(private route: ActivatedRoute) {
   route.params.subscribe((params) => {
     this.courseId = params['id'];
   });
}
```

Origem:

```
<a routerLinkActive="active"
  [routerLink]="['course', courseId]">
    Detalhes  Curso
</a>
```

Navegação via código

- Utilizamos a classe Router

```
constructor(private router: Router) { }
```

```
this.router.navigate(['/event', 5]);
```

Criando rotas Filhas

- São usadas quando o objetivo é exibir mais de um componente na mesma rota, reutilizando o componente pai

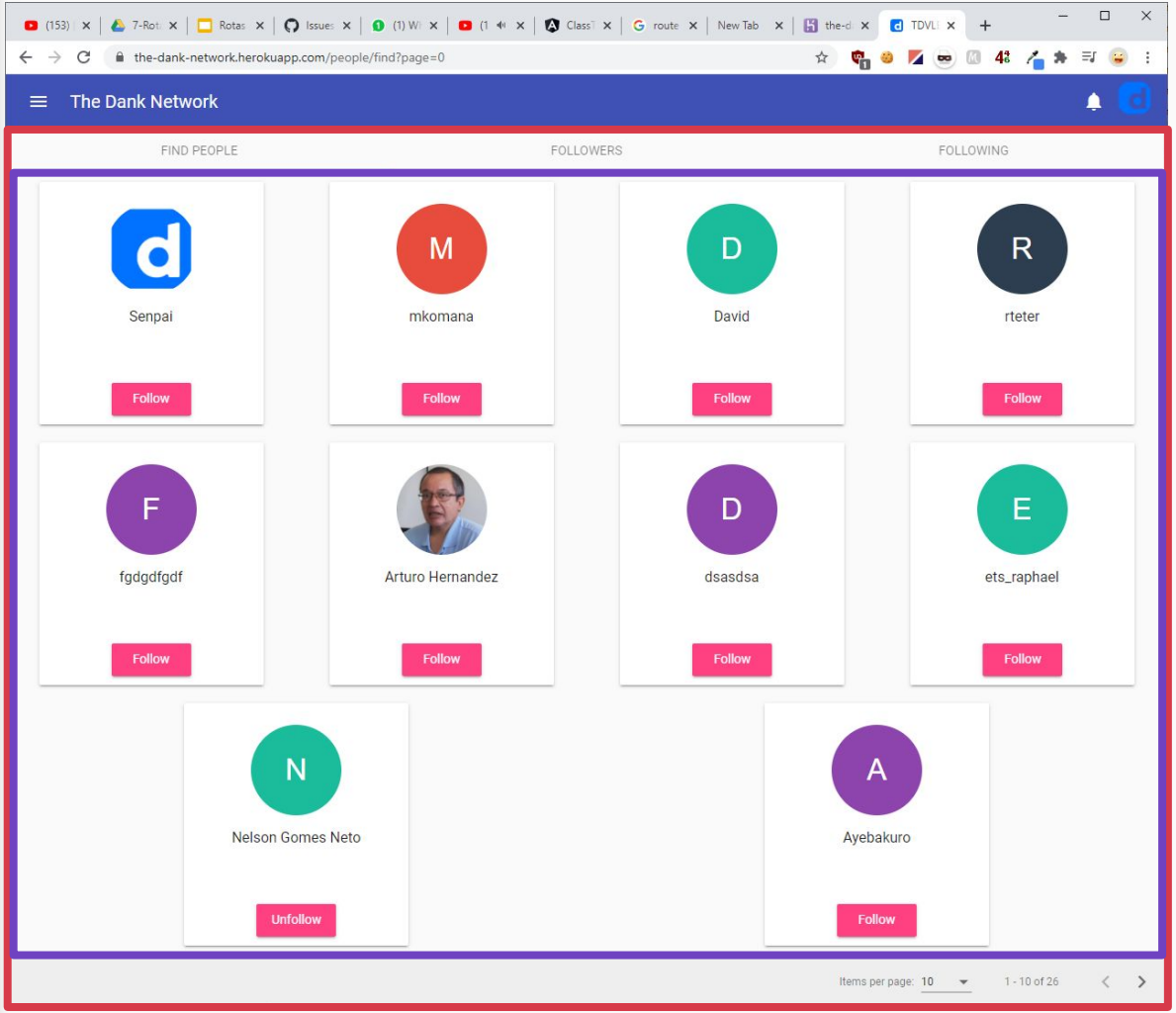
Rotas filhas

- Para utilizar mais de um componente na mesma tela, na declaração das rotas, além do path e do componente, declarar também os filhos:

```
...
    {path: 'students', component:
    StudentsComponent, children: [
        {path: novo, component:
    StudentFormComponent}
    ]}
...
```

- Adicionar a tag <router-outlet></router-outlet> no componente pai

Rotas Filhas



Rota parent: people

Rotas filhas: find people, followers e following

Exemplo: rotas filhas

- No caso do exemplo anterior, a declaração de rotas seria algo como:

```
...
    {path: 'people', component: PeopleComponent, children: [
        {path: 'find', component: FindPeopleComponent},
        {path: 'followers', component: FollowersComponent},
        {path: 'following', component: FollowingComponent}
    ]}
...
```

- E no PeopleComponent seria necessário um <router-outlet> para informar ao Angular em qual local do template serão renderizados os componentes filhos

Lazy Loading

- Padrão de projeto de software
- “Adia a inicialização de um objeto até o ponto em que ele é necessário, contribuindo para a eficiência no funcionamento de um programa.”
- Vantagens:
  - Ganho de performance
  - Segurança

Lazy Loading

1. Organizar a aplicação em módulos
2. Dentro do arquivo principal de rotas da aplicação utilizar o loadChildren
3. Remover qualquer importação do módulo utilizado no loadChildren
4. No arquivo de roteamento do módulo, o caminho principal deve ficar vazio

Obs.: *reiniciar o ng serve*

Guarda de Rotas

- Permitem ou não o acesso a determinadas rotas

  ng g guard guard-name

- São serviços que implementam uma interface de guarda de rota
  - CanActivate
  - CanActivateChild
  - CanDeactivate
  - Resolve
  - CanLoad
- Referência: https://angular.io/guide/router#preventing-unauthorized-access

Definindo rota padrão

- Utilizar path com valor *'**'*

```
{path: '**', redirectTo: 'login', pathMatch: "full"}
```

- pathMatch: maneira de indicar como a verificação da rota deve ser feita
  - full
  - prefix

Referência: https://angular.io/guide/router

Observações:

- Questionário: https://forms.gle/3kPAR8Sfq2PUJnSN9
- Código fonte: https://github.com/GabriellBP/Angular-course
- Dúvidas:
  - gabriel.pereira@edge.ufal.br
  - lucas.amorim@edge.ufal.br

Obrigado!