

Workshop n°4 :

Manipulation des modules & Lazy Loading

Les objectifs :

- Créer et organiser des modules fonctionnels (Feature Module).
- Générer et structurer des composants à l'intérieur d'un module.
- Comprendre la relation déclarations / imports / exports dans un NgModule.
- Mettre en place un routage interne avec lazy loading.
- Gérer une première route paramétrée avec un composant DetailEvent.

Les instructions à suivre :

- 1- Créer la liste des modules fonctionnels

Taper la commande pour créer le module Events avec le lazy loading:

ng g module features/events --route events --module app.module

Cette commande permet de :

- Créer `events.module.ts`
- Créer `events-routing.module.ts`
- Ajouter une route lazy dans le fichier `app.routing.ts /events`.

ng g module features/tickets --route tickets --module app.module

ng g module features/feedback --route feedback --module app.module

ng g module features/users --route users --module app.module

```
const routes: Routes = [  
  {path: '', component: HomeComponent},  
  {path: 'home', redirectTo: '', pathMatch: 'full'},  
  { path: 'events', loadChildren: () => import('./features/events/events.module').then(m => m.EventsModule) },  
  { path: 'tickets', loadChildren: () => import('./features/tickets/tickets.module').then(m => m.TicketsModule) },  
  { path: 'feedback', loadChildren: () => import('./features/feedback/feedback.module').then(m => m.FeedbackModule) },  
  { path: 'users', loadChildren: () => import('./features/users/users.module').then(m => m.UsersModule) },  
  {path: '**', component: NotFoundComponent},  
];
```

Figure: le fichier `app.routing.module.ts` avec le lazy loading

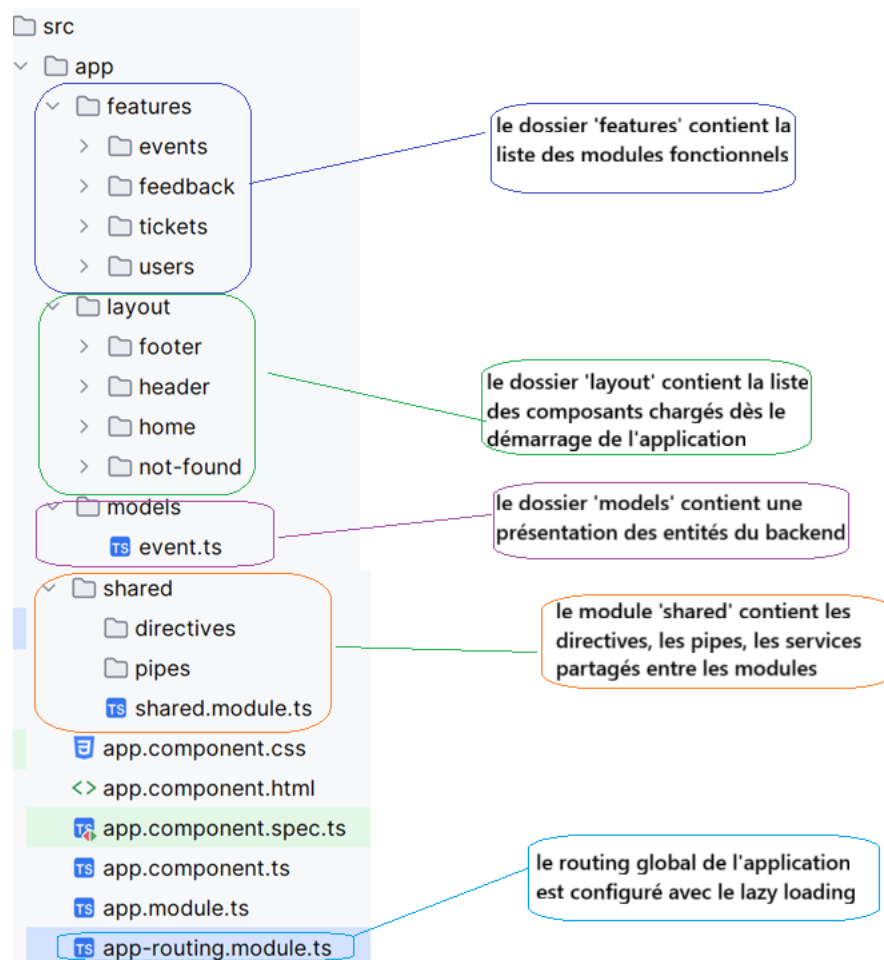


Figure : Structure du projet

2- Créer des composants pour le module Event :

- Placer le composant 'List-Events' sous le dossier pages de module 'events' s'il est déjà créé et compléter les imports nécessaires pour assurer le fonctionnement de ce composant.
- Créer les composants suivants :

ng g c features/events/ event-detail --skip-tests

ng g c features/events/ event-card --skip-tests

ng g c features/events/ /search-bar --skip-tests

- **Event-Detail** : affiche les détails d'un évènement.
- **Search-bar**: une barre de recherche par adresse, prix, date ...
- **EventCardComponent** : composant réutilisable pour afficher un événement.

3- Routage interne du module Events

- Configurer le fichier `events-routing.module.ts` en ajoutant deux routes :
 - La route `/events` qui charge le composant **ListEventComponent**.
 - La route `/events/:id` (exemple `/events/2`) qui charge le composant **EventDetailComponent**.

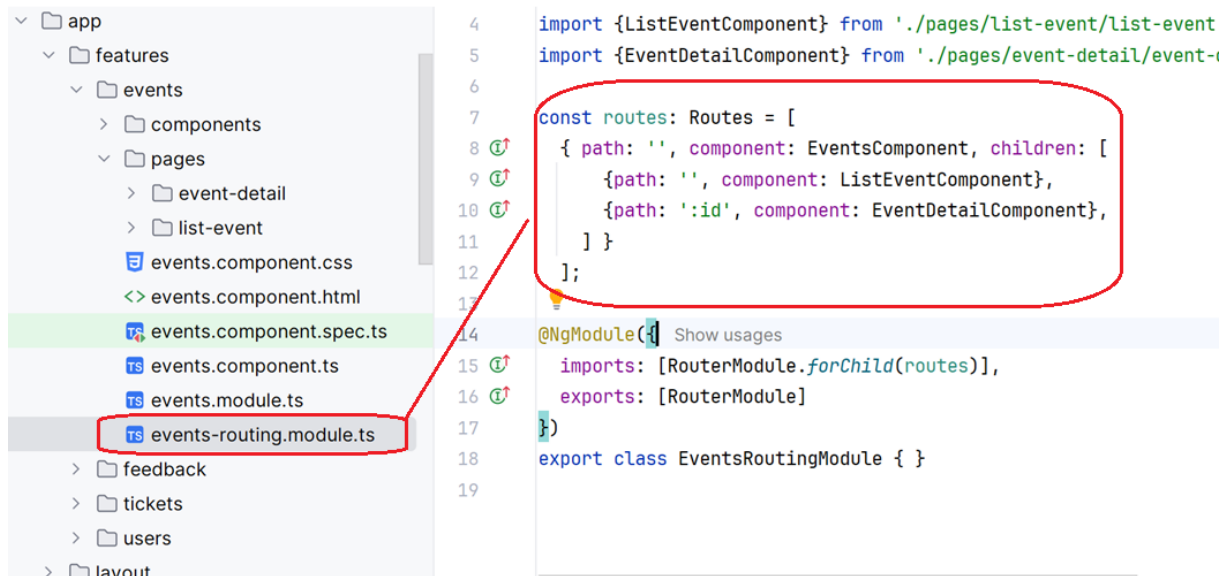


Figure : le fichier de routing interne du module events

- b. Ajouter la balise `<router-outlet></router-outlet>` dans le template du composant **events.component.html**, qui sera considéré comme le composant racine du module *Events*.

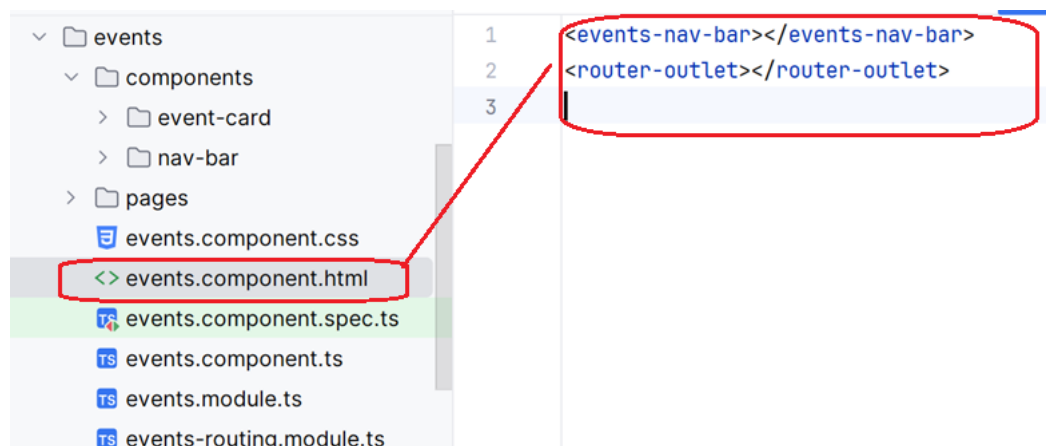


Figure : le fichier html du composant racine

4- Configurer une route paramétrée :

- a. Dans le fichier html du composant List-event ajouter un lien paramétré permettant le chargement du composant Event-Detail:

```

<button class="btn btn-sm btn-primary"
  (click)="like(e)"
  [disabled]="isExpired(e) || e.nbPlaces <= 0">
  Like
</button>
<a [routerLink]="['./',e.id]">see details</a>

```

- b. Dans le composant 'EventDetail' afficher les détails de l'évènement sélectionné :
 - i. Utiliser le service 'Activated Route' pour récupérer l'id de l'évènement
 - ii. Créer un service pour partager les données (list-events) sous un dossier 'data-access'

Explication :

| Commande | Rôle |
|---|--|
| ng g module features/events --route events --module app.module | Créer le module Events avec lazy loading |
| ng g module features/tickets --route tickets --module app.module | Créer le module Tickets avec lazy loading |
| ng g c features/events/pages/event-list --skip-tests | Créer le composant EventList |
| ng g c features/events/pages/event-detail --skip-tests | Créer le composant EventDetail |
| ng g c features/events/components/event-card --skip-tests | Créer le composant EventCard |
| ng g c features/events/components/search-bar --skip-tests | Créer un composant de recherche réutilisable |
| ng g s data-access/events --skip-tests | Créer le service events |

Ressources sont disponibles : [lien](#)

<https://github.com/badi3a/AngularTraining/tree/workshop-04-ManipulateModule>