

Workshop n°3 :

Manipulation des Composants

Les objectifs :

- Créer un composant Angular (ListEvent) dans l'AppModule.
- Manipuler les différents types de data binding :
 - o Interpolation ({{ }})
 - o Property binding ([property])
 - o Event binding ((event))
 - o Two-way binding ([ngModel])
- Afficher une liste statique d'événements à partir d'un modèle.
- Mettre à jour dynamiquement les données (incrémenter un like).
- Gérer les états (désactiver un bouton pour un événement expiré).
- Implémenter une recherche d'événements avec ngModel.

Le rendu final :

- Création d'un composants fonctionnels : ListEvent
- Création de l'interface « ListEvent »
- Configurer le routage vers l'interface « ListEvent »

Les instructions à suivre :

- 1- Générer le composant List-event :
Taper la commande: `ng g c layout/listEvent --skip-tests`
- 2- Générer l'interface Event:
Taper la commande: `ng g i models/event`

```
// We use "export" so that this interface can be imported and reused
//   in other components, services, or modules of the application.
export interface Event { no usages
  // We use "interface" instead of "class" here because:
  //   - It defines only the *shape* (structure) of the data.
  //   - It does not generate JavaScript code (lighter).
  //   - It is perfect for describing data models coming from an API or static data
  id: number;           // Unique identifier of the event
  titre: string;        // Event title
  description: string; // Event description
  date: Date;           // Event date
  lieu: string;          // Location
  prix: number;          // Ticket price
  organisateurId: number; // Organizer identifier
  imageUrl: string;     // Path/URL to the event image
  nbPlaces: number;      // Number of available places
  nbrLike: number;       // Number of likes
}
```

Figure : l'interface Event

- Ensuite, créer une liste des évènements statique dans le composant « ListEvent »
- 3- Créer une liste statique d'événements dans le composant ListEvent.
 - 4- Développer l'interface utilisateur pour afficher la liste des événements.
 - 5- Configurer un lien vers la liste des événements dans la barre de navigation et sur la page d'accueil afin de rediriger l'utilisateur vers l'interface ListEvent.
 - 6- Lors d'un clic sur le bouton Like, le nombre de likes de l'événement doit s'incrémenter.
 - 7- Le bouton Like doit être désactivé pour les événements expirés.
 - 8- Ajouter une barre de recherche permettant de filtrer les événements par titre et lieu..

The screenshot shows the 'EventHub' application's 'Liste des événements' (Events list) page. At the top, there is a navigation bar with links for 'Accueil', 'Événements', 'Mes Tickets', 'Feedback', and 'Se connecter'. Below the navigation bar, a search bar contains the placeholder 'Rechercher un événement'. The main content area displays two event cards:

- Angular Summit**: Located in Tunis on Nov 10, 2025. It's a conference on Angular and front-end ecosystem. Price is TND50.000 and 25 places are available. It has 4 likes.
- Web Dev Days**: Located in Ariana on Jan 5, 2025. It's a day dedicated to modern web frameworks. Price is TND30.000 and no places are available. It has 3 likes.

At the bottom of the page is a footer with the 'EventHub' logo, a tagline 'Plateforme de gestion d'événements, billetterie en ligne et feedback.', and links for 'Liens', 'Événements', 'Mes Tickets', 'Feedback', and 'Se connecter'. It also includes contact information: 'Email : contact@eventhub.tn' and 'Site : eventhub.tn'.

Figure: L'interface liste des Evènements

Commande	Rôle
ng serve -open	Lancer le projet (ouvre le navigateur automatiquement)
ng s -o npm start	
ng g i models/event	Créer une interface sous le dossier models
ng g c layout/listEvent --skip-tests	Créer le composant ListEvent sous le dossier layout et ignorer les fichier spec

Demo disponible : <https://event-hub-one-alpha.vercel.app>

Ressources sont disponibles : [lien](#)

<https://github.com/badi3a/AngularTraining/tree/workshop-03-manipulate-component>