


	Année Universitaire : 2023-2024 
<p>Workshop n°3 :</p> <h2 style="text-align: center;">Manipulation de composant</h2>	

Objectifs :

- Créer et manipuler un composant
- Appliquer le databinding (event-binding, property-binding, two-way databinding)
- Définir un modèle de données
- Manipuler les directives NgFor, ngIf, ngModel

Etude de cas:

Créer une SPA (Single Page Application) permettant à l'utilisateur de:

- Parcourir une liste des utilisateurs
- Supprimer un utilisateur
- Chercher un utilisateur selon le nom.

Travail à faire :

1. Créer un projet Angular et créer l'interface suivante comme page d'accueil si vous ne l'avez pas déjà (travail demandé dans le workshop 1)



Figure 1

2. Créer un composant appelé **ListUserComponent**.
3. Afficher le composant créé dans la page d'accueil.
4. Créer un modèle de données appelé «User » à définir sous un répertoire model comme suit :

```
export class User{
  idCustomer: number;
  firstName: string;
  lastName: string;
  birthDate: string;
  email: string;
  password: string;
  profession: string;
  accountCategory: string;
  picture: string;
}
```

5. Ajouter la liste des users suivants dans la propriété « list » du composant ListUserComponent

```
[
  {
    idCustomer: 1,
    firstName: "Mila",
    lastName: "Kunis",
    birthDate: "1999-06-30",
    accountCategory: "Admin",
    email: "mila@kunis.com",
    password: "test",
    picture: "https://bootdey.com/img/Content/avatar/avatar3.png",
    profession: "Software Engineer"
  },
  {
    idCustomer: 2,
    firstName: "George",
    lastName: "Clooney",
    birthDate: "1999-06-30",
    accountCategory: "Customer",
    email: "marlon@brando.com",
    password: "test",
    picture: "https://bootdey.com/img/Content/avatar/avatar2.png",
    profession: "Software Engineer"
  },
  {
    idCustomer: 3,
    firstName: "George",
    lastName: "Clooney",
    birthDate: "1999-06-30",
    accountCategory: "Customer",
    email: "marlon@brando.com",
    password: "test",
    picture: "https://bootdey.com/img/Content/avatar/avatar1.png",
  }
]
```

```

    profession: "Software Engineer"
  },
  {
    idCustomer: 4,
    firstName: "Ryan",
    lastName: "Gossling",
    birthDate: "1999-06-30",
    accountCategory: "Golden",
    email: "Ryan@nicholson.com",
    password: "test",
    picture: "https://bootdey.com/img/Content/avatar/avatar4.png",
    profession: "Software Engineer"
  },
  {
    idCustomer: 5,
    firstName: "Robert",
    lastName: "Downey",
    birthDate: "1999-06-30",
    accountCategory: "Blocked Account",
    email: "robert@nicholson.com",
    password: "test",
    picture: "https://bootdey.com/img/Content/avatar/avatar5.png",
    profession: "Software Engineer"
  }
]

```

6. Afficher la liste des utilisateurs définis dans la propriété **list** du composant créé comme dans la Figure 2. Le css et html de la liste user sont disponibles sur:

bootsnipp : <https://www.bootdey.com/snippets/view/User-list-page>
 avatar user: <https://www.bootdey.com/snippets/view/Box-gallery-images>



User Name	Born Day	Profession	Email	
 John Smith Admin	01-01-1980	Software Engineer	john@mystore.com	delete
 George Cooper Customer	01-01-1981	Software Engineer	george@mystore.com	delete
 Ryan Cassiano Admin	01-01-1981	Software Engineer	ryan@mystore.com	delete
 Robert Cooper Blocked Account	01-01-2000	Software Engineer	robert@mystore.com	delete

Figure 2: L'interface finale

7. Ajouter le bouton « Delete » devant chaque user.
8. Le bouton « Delete » **est inactif** si la catégorie du user égale 'Admin'.
9. Le clique sur le bouton « Delete » permet de supprimer l'utilisateur sélectionné.
10. Ajouter un input de type « text » qui permet à l'utilisateur de chercher des utilisateurs selon la catégorie.