|  |  |
| --- | --- |
|  | Année Universitaire : 2023-2024 |
| **Workshop n°4 :**  **Création et Validation de Formulaire**  **« Reactive Form »** | |

# Objectifs :

* Créer un formulaire avec l’approche Reactive Form
* Ajouter des validateurs et les messages d’erreurs adéquats

# Etude de cas :

Nous souhaiterions créer une application pour une agence immobilière spécialisée dans la vente des appartements en utilisant le framework Angular côté FrontEnd. Sachant que la partie opérationnelle (la couche métier) est déjà développée et elle est exposée sous forme des services web côté backend.

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

a. Parcourir une liste de résidences.

b. Parcourir une liste d’appartements selon une résidence sélectionnée.

c. Acheter un appartement.

d. Chercher un appartement par surface.

# Prérequis

# Projet Angular crées selon l’arborescence définie dans le workshop n°2

# Le modèle de données « Apartment » déjà créé sous le répertoire models.

# x

# Travail à faire :

1. Au niveau du composant **FormAppartmentComponent (Créé dans le workshop 3)**, créer une propriété appelée « **apartForm** » de type **FormGroup** qui correspond au formulaire de la Figure 1 et qui permet d’ajouter un appartement. Chaque élément du formulaire est de type **FormControl**

NB : N’oubliez pas d’importer le module « ReactiveFormsModule » dans le module racine.

|  |
| --- |
|  |

Figure 1: Formulaire d'ajout d'un appartement

NB : Le champ Residence correspond à l’id de la résidence passée en paramètre via l’url

1. Voici les validateurs à appliquer sur ce formulaire :

* Tous les champs sont obligatoires
* Les champ Apartment Number et Floor Number n’acceptent que des chiffres.
* Le champ description contient au moins 10 caractères
* La Surface Terrace n’est accessible que si le champ Terrace est coché “yes”

1. Au niveau du template du composant, créer le HTML du formulaire de la figure 1 et lier chaque élément html avec son équivalent de la propriété **apartForm** grâce aux directives **FormGroup** et **FormControlName**.
2. Ajouter pour chaque champ le message d’erreur adéquat en cas de non-respect du validateur correspondant.
3. Le bouton « Add » est désactivé tant qu’il existe au moins un champ du formulaire invalide.