# TD n°2


- Réalisation du TD n°1
- Directives
- Services
- Filtres





## -- Exercice : choix multiples

### -- Objectifs

1. Créer un module et un contrôleur
2. Utiliser des directives Angular
3. Mettre en oeuvre le Data-binding

### -- Fonctionnalités

1. Lister les produits disponibles
2. Ajouter un ou plusieurs produits disponibles aux produits sélectionnés
3. Ajouter tous les produits disponibles aux produits sélectionnés
4. Retirer un ou plusieurs produits sélectionnés
5. Retirer tous les produits sélectionnés
6. Valider le choix de produits sélectionnés
7. Revenir au choix des produits

### -- Application/Contrôleurs

| | |
|---|---|
| **Application** | **ListesApp** (app/listesApp.js) |
| **Contrôleur** | **ListesController** (app/listesController.js) |
| | dispoItems<br>Tableau des produits disponibles défini en JSON (utiliser le service $http) |
| | includedItems<br>Tableau des produits à inclure |
| | selectedDispoItems<br>Tableau des éléments sélectionnés dans la liste des produits disponibles |
| | selectedIncludedItems<br>Tableau des éléments sélectionnés dans la liste des produits à inclure |
| | step<br>Variable d'état agissant sur les vues et prenant les valeurs 1 : choix des produits ou 2 : visualisation du choix |
| | addToIncluded()<br>Ajoute les produits disponibles en surbrillance (**selectedDispoItems**) à la liste des produits à inclure (**includedItems**) |
| | addAllToIncluded()<br>Ajoute tous les produits disponibles (**dispoItems**) à la liste des produits à inclure (**includedItems**) |
| | removeFromIncluded()<br>Retire les produits sélectionnés en surbrillance (**selectedIncludedItems**) pour les remettre dans |

**dispoItems** : à intégrer dans le contrôleur

```
[
        {
            "url":
"http://tutorialzine.com/2013/07/50-must-have-plugins-for-extending-twitter-bootstr
ap/",
            "title": "50 Must-have plugins for extending Twitter Bootstrap",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/07/featured_4-100x100.jpg"
        },
        {
            "url":
"http://tutorialzine.com/2013/08/simple-registration-system-php-mysql/",
            "title": "Making a Super Simple Registration System With PHP and
MySQL",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/08/simple_registration_system-
100x100.jpg"
        },
        {
            "url": "http://tutorialzine.com/2013/08/slideout-footer-css/",
            "title": "Create a slide-out footer with this neat z-index trick",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/08/slide-out-footer-100x100.jp
g"
        },
        {
            "url": "http://tutorialzine.com/2013/06/digital-clock/",
            "title": "How to Make a Digital Clock with jQuery and CSS3",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/06/digital_clock-100x100.jpg"
        },
        {
            "url": "http://tutorialzine.com/2013/05/diagonal-fade-gallery/",
            "title": "Smooth Diagonal Fade Gallery with CSS3 Transitions",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/05/featured-100x100.jpg"
        },
        {
            "url": "http://tutorialzine.com/2013/05/mini-ajax-file-upload-form/",
            "title": "Mini AJAX File Upload Form",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/05/ajax-file-upload-form-100x1
00.jpg"
        },
        {
            "url": "http://tutorialzine.com/2013/04/services-chooser-backbone-js/",
            "title": "Your First Backbone.js App – Service Chooser",
            "image":
"http://cdn.tutorialzine.com/wp-content/uploads/2013/04/service_chooser_form-100x10
0.jpg"
        }
    ]
```

## -- Vues

### -- Directives utilisées

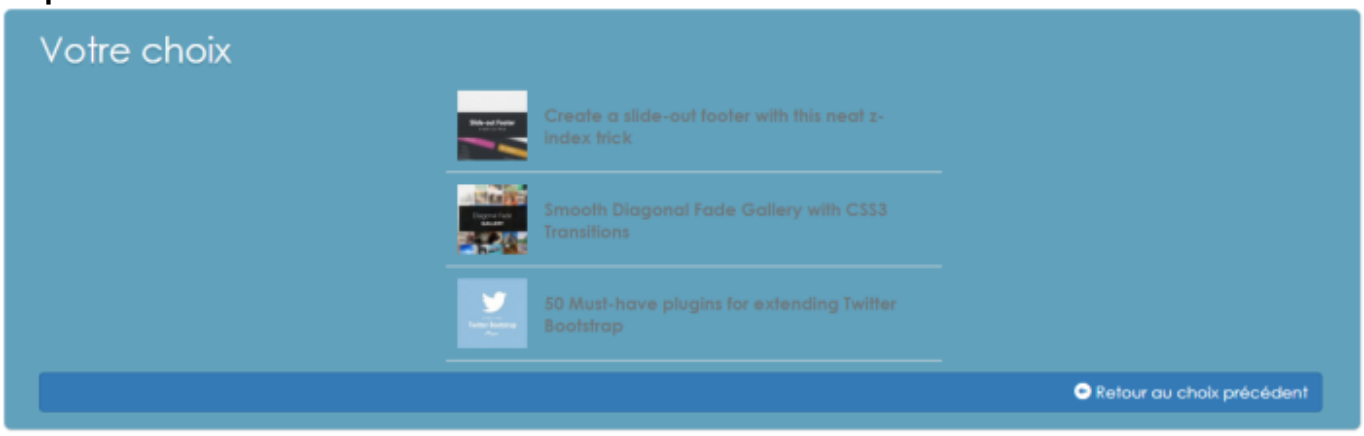
- ng-app
- ng-controller
- ng-repeat
- ng-options
- ng-pluralize
- ng-click
- ng-dblclick
- ng-src
- Expressions {{expression}}

### -- Interface

**step==1 :**




**step==2 :**




### -- Logique applicative / comportement de l'interface

- Les 4 boutons d'ajout ou de retrait des services à inclure s'activent ou se désactive (disabled) en fonction du contenu des listes et/ou des éléments sélectionnés
  - bouton ▶| : ajoute le(s) élément(s) sélectionnés de la liste de gauche à la liste de droite
  - bouton ▶▶| : ajoute tous les éléments sélectionnés de la liste de gauche à la liste de droite
  - bouton |◀ : retire le(s) élément(s) sélectionnés de la liste de droite et les place dans la liste de

gauche
- bouton |◀◀ : retire tous les éléments sélectionnés de la liste de droite et les place dans la liste de gauche

- Un double clic sur l'un des éléments des 2 listes ajoute ou retire l'élément et le passe dans la liste associée
- Le nombre de produits inclus se met à jour en fonction du nombre d'éléments dans la liste de droite

- Le passage à l'étape suivante n'est possible que si la liste de droite n'est pas vide
  - le bouton “Passer à l'étape suivante” affiche la vue suivante (en passant step à 2)
  - la vue affiche la liste des éléments ajoutés dans les éléments à inclure

# -- Exercice : Gestion des contacts

## -- Objectifs

1. Utiliser la directive ngRepeat
2. Utiliser la validation des formulaires
3. Créer des directives simples (voir Création de directives)

## -- Fonctionnalités

1. Obtenir la liste des contacts et la filtrer
2. Ajouter un contact dans la liste
3. Modifier un contact existant
4. Supprimer un contact

## -- Application/Contrôleurs

| | |
|---|---|
| **Application** | **ContactApp** (app/ContactApp.js) |
| **Contrôleur** | **ContactController** (app/contactController.js) |
| | contacts<br>Tableau des contacts existants |
| | contact<br>Variable contenant le contact à modifier |
| | tmpContact<br>Variable contenant le contact modifié par le formulaire |
| | operation<br>Chaîne indiquant la **modification** ou l'**ajout d'un contact** |
| | edit<br>Contrôle l'affichage du formulaire d'ajout/modification |
| | toUpdate(contact)<br>Affiche le formulaire de modification du contact |
| | toAdd()<br>Affiche le formulaire d'ajout d'un contact |
| | add()<br>Ajoute le contact |
| | update()<br>Met à jour la liste des contacts après validation du formulaire (ajout ou modif) |
| | delete(contact)<br>Supprime de la liste le contact |

## -- Vue

### -- Directives utilisées

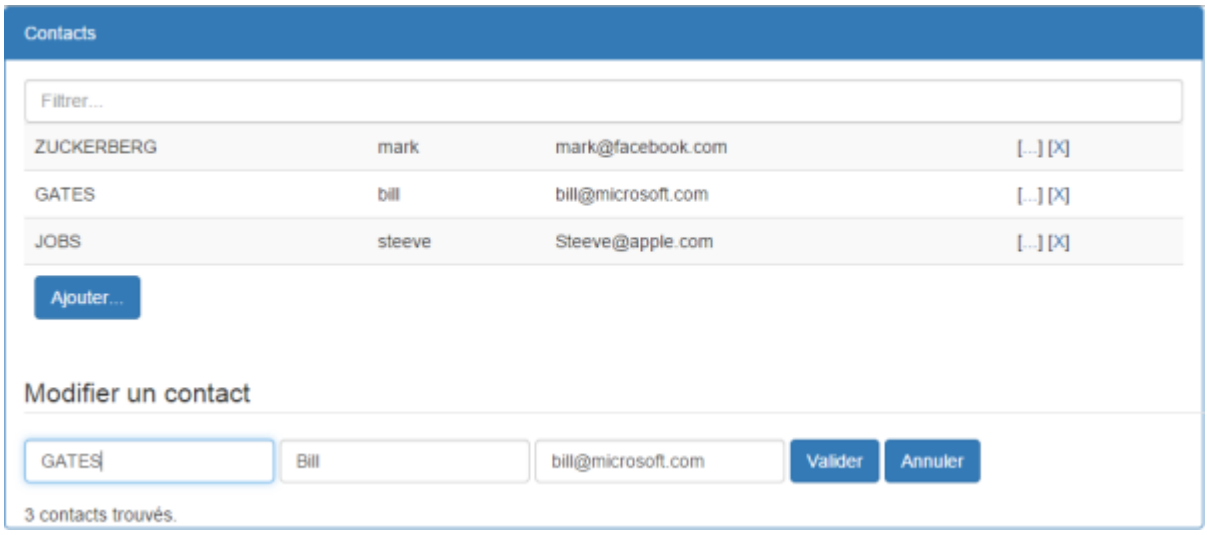
- ng-app
- ng-controller
- ng-click
- ng-model
- ng-pluralize
- ng-repeat

### -- Directives à créer

A placer dans **app/directives.js**

| Nom | Description |
|---|---|
| **contactElem** | Permet d'afficher un contact sur une ligne de tableau |
| **frmContact** | Affiche un formulaire de contact |

### -- Interface




Tester l'appli **Gestion de contacts**

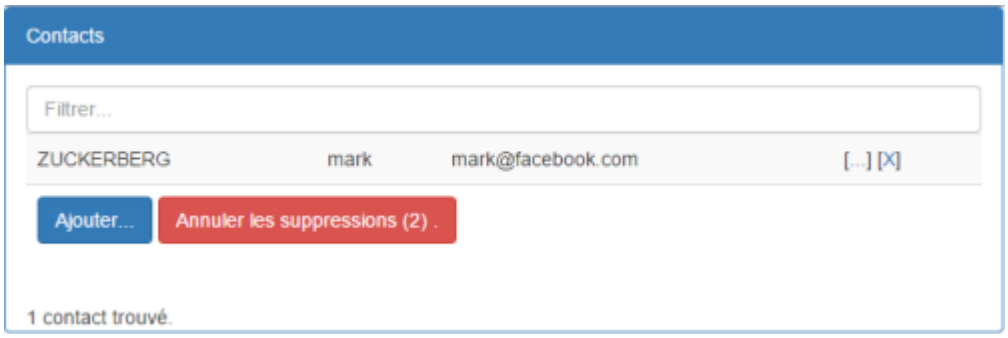
### -- Logique applicative / comportement de l'interface

- Zone filter : filtre l'ensemble des contacts affichés.
- Boutons :
  - **[x]** : supprime le contact associé
  - **[...]** : affiche le formulaire de modification du contact
  - **Ajouter** : affiche le formulaire d'ajout de contact
- Formulaire d'ajout/modification
  - Le champ **nom** est obligatoire
  - Le champ **mail** doit comporter une adresse email valide
  - Le bouton **Valider** ajoute le contact à la liste des contacts et masque le formulaire
  - Le bouton **Annuler** masque le formulaire

**-- Création d'un filtre**

Variante sur la suppression permettant l'annulation d'opérations :

1. Créer un filtre dans le module nommé **notDeleted** permettant de retourner, parmi un tableau de contacts, ceux qui ne sont pas marqués comme supprimés : **contact.deleted=true**
2. Appliquer le filtre à l'affichage de la liste de contacts
3. Modifier la méthode delete du contrôleur
4. Ajouter un bouton Annuler la ou les suppressions (visible s'il existe un élément supprimer) et affichant le nombre de suppressions à annuler, et la méthode **cancelOneDeletion()**




From:
http://slamwiki2.kobject.net/ - **SlamWiki 2.1**

Permanent link:
**http://slamwiki2.kobject.net/slam4/richclient/angularjs/td2?rev=1487061213**

Last update: **2019/08/31 14:39**