TD n°3



Tous les exercices utilisent Bootstrap pour la partie CSS.
`ember install ember-bootstrap`

Pour passer à la v4 de bootstrap (sass requis) :
`npm install --save-dev ember-cli-sass`
`ember generate ember-bootstrap --bootstrap-version=4`

pour les fonts :
`ember install ember-font-awesome`

- Bases emberJS
- Computed properties
- Actions
- Models
- Opérations CRUD

Renommer le projet **td2** en **td2-3**

Ajouter le plugin **ember-localstorage-adapter** pour stocker des models côté client :

```
ember install ember-localstorage-adapter
```

Adapter et Serializer

Créer l'adapter :

```
ember g adapter application
```

Définir le namespace **contact-app**, clé sous laquelle les données seront enregistrées dans le local storage

```
import LSAdapter from 'ember-localstorage-adapter';

export default LSAdapter.extend({
  namespace: 'contact-app'
});
```

Créer le serializer :

```
ember g serializer application
```

```
import { LSSerializer } from 'ember-localstorage-adapter';

export default LSSerializer.extend();
```

Gestion des contacts

-- Objectifs

1. Models et Store
2. Computed properties
3. Création de composants

- 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
5. Annuler la suppression

Model

	contact
Model	nom string
	prenom string
	email string

-- Route/Model/actions

/contacts

Affiche la liste des contacts

	Contacts
Classe	datas Tableau des contacts provenant du store
	contacts Liste filtrée des contacts non supprimés (* sur filtre et datas.@each.isDeleted)
	deleteds Liste des contacts supprimés
	deletedsCount Nombre de contacts supprimés
Route handler	model() Retourne une instance de Contacts initialisée avec le **datas** des contacts provenant du store
actions	delete(contact) Supprime le contact (sans sauvegarde dans le store)
	cancelDeletion(deleteds) Annule la suppression de tous les contacts supprimés

/contacts/add

Affiche le formulaire d'ajout d'un contact

Route handler	model() Retourne l'instance de contact à modifier
actions	addContact(datas) Ajoute un nouveau contact dans le store

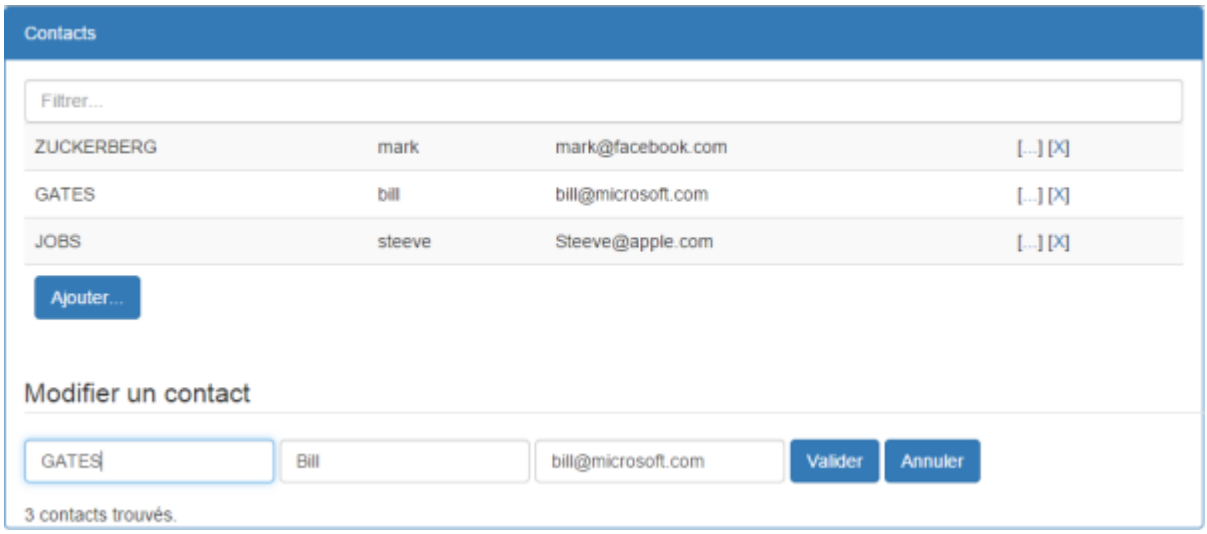
/contacts/edit

Affiche le formulaire de modification d'un contact

Route handler	model() Retourne l'instance de contact à modifier
actions	updateContact(datas) Modifie le contact dans le store

- Vues

- 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

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

Permanent link:
http://slamwiki2.kobject.net/richclient/emberjs/td3?rev=1551021779

Last update: **2019/08/31 14:38**