

TPN °1 :

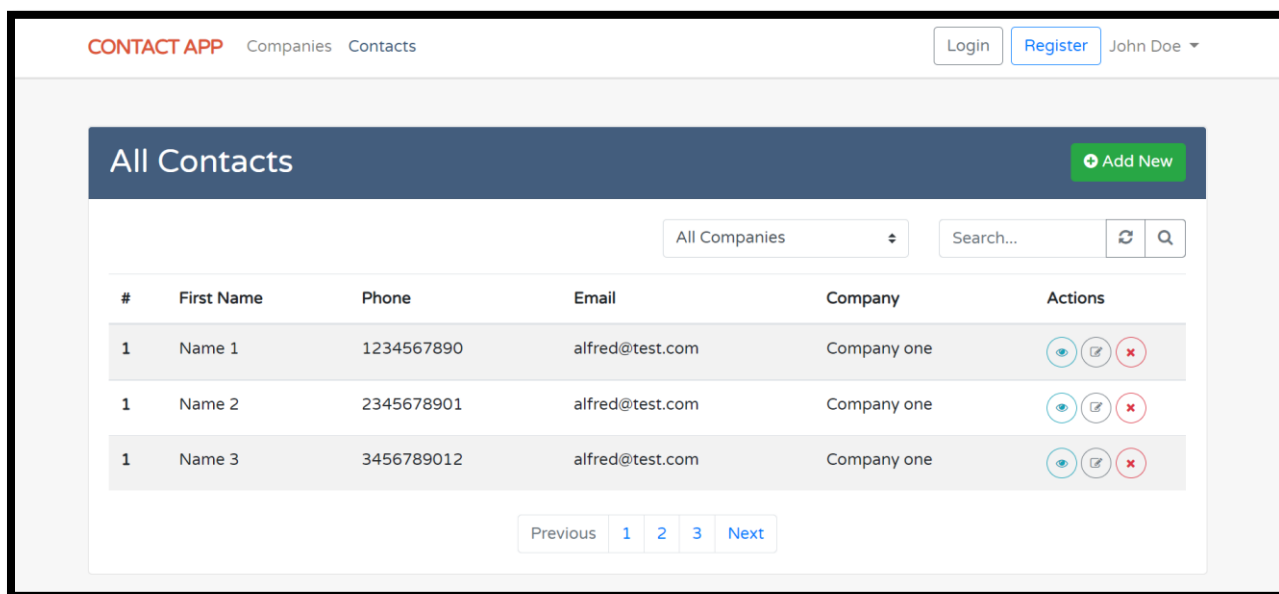
Laravel Routes + View

OBJECTIFS :

- *Pratiquer les principes des routes et views*
- Paramètres de routage
- Routes nommées
- Groupe de routage
- Affichage des données dans le Blade

Travail demandé :

Dans cette partie, nous allons intégrer ce modèle HTML dans notre application.



- 1- Créer un projet laravel : **Contact-App**.
- 2- Allez aux ressources, répertoire view, créer un nouveau répertoire nommé « **layouts** », Dans ce dossier, créons un nouveau fichier. Nommé le main.blade.php . dans ce fichier, créez le NavBar de l'image ci-dessus. (utiliser bootstrap)
- 3- Dans le dossier ressources, créez un répertoire « **contacts** », qui contient 3 fichiers suivant :
 - create.blade.php
 - index.blade.php

- show.blade.php

Dans les 3 fichiers, appeler le navbar crée dans le fichier **main.blade.php**. utiliser « extends » .

4- Criez 3 routes :

- /contacts : redirige vers la view « index », et nous passe la liste des contacts vers la view.

Utiliser une fonction **getContacts** qui nous retourne la liste des contacts

Nommé ce route par « contacts.index »

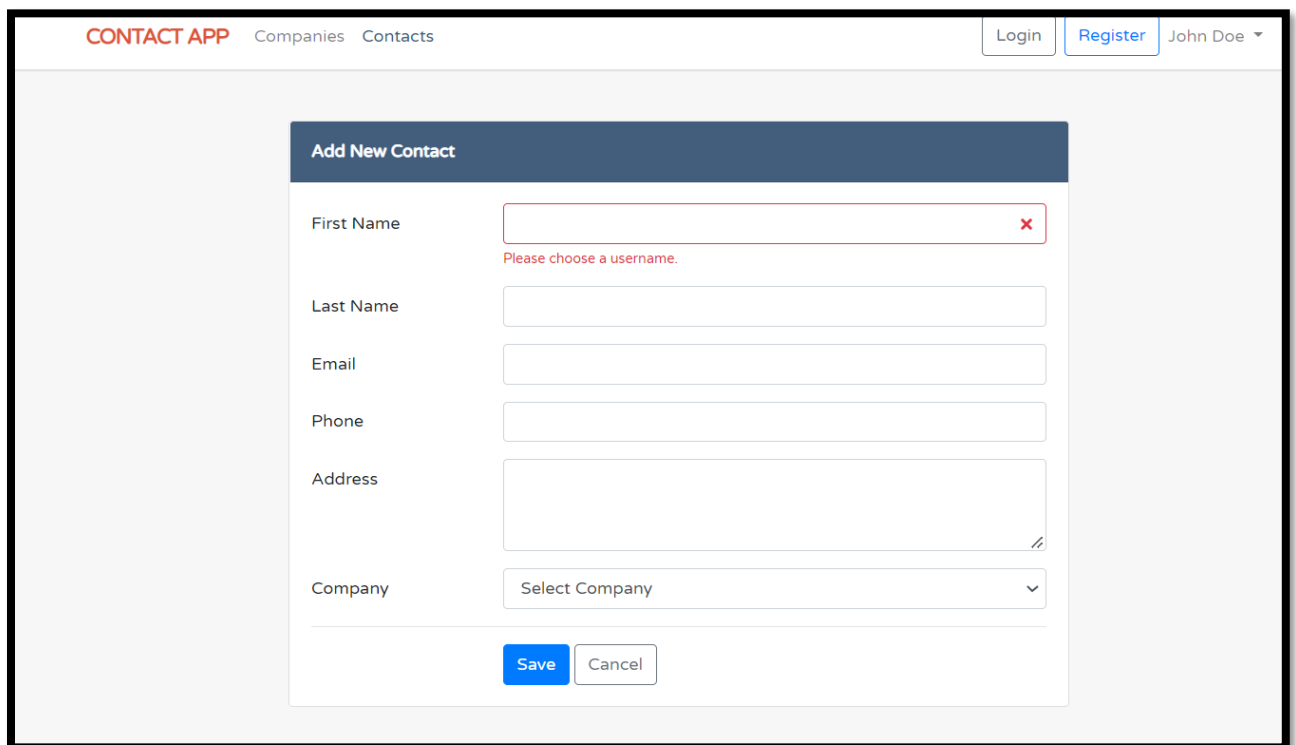
Afficher les contacts sur la view index comme indiqué sur l'image ci-dessus.

Le tableau doit être crée dans une section « **content** »

- /contacts/create : redirige vers la view « create », la view create contient la formulaire suivante :

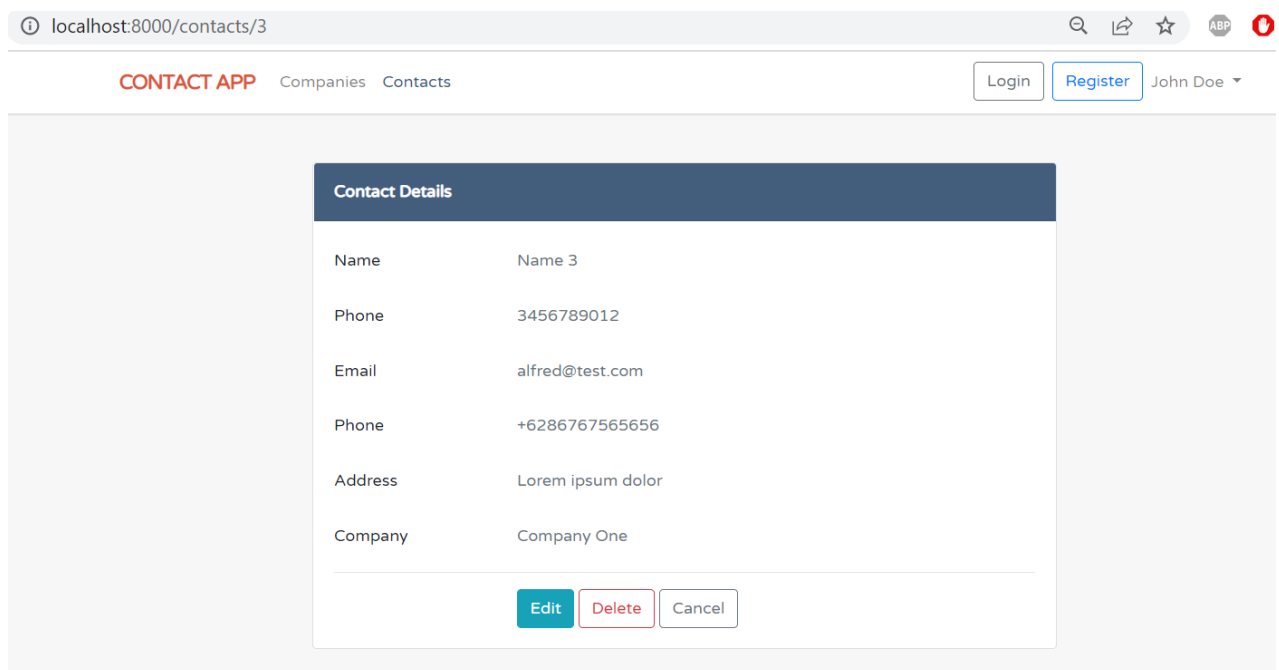
Nommé cette route par « contacts.create »

Le formulaire doit être crée dans une section « **content** »



The screenshot shows a web application interface for a 'CONTACT APP'. The top navigation bar includes 'Companies' and 'Contacts' links, along with 'Login' and 'Register' buttons, and a user profile 'John Doe'. The main content area displays a modal form titled 'Add New Contact'. The form contains the following fields: 'First Name' (with a red border and an error message 'Please choose a username.'), 'Last Name', 'Email', 'Phone', 'Address' (with a text area and a small icon), and 'Company' (a dropdown menu with 'Select Company' as the placeholder). At the bottom of the form are 'Save' and 'Cancel' buttons.

- `/contacts/{id}` : redirige vers la view « show », qui prend l'id d'un contacte en paramètre, il retourne le contacte s'il est trouvé, sinon redirige vers une page « Not found »
- Nommé ce route par « `contacts.show` »
- Le détail du contacte doit être crée dans une section « **content** »



- 5- Déplacez certaines fonctionnalités dans des **sous-vues** et utilisez la directive **include** :
- Créez deux view : `_company-selection.blade.php` et `_filter.blade.php`
 - Sur la route `/contacts`, créez une liste des **companies** et le passer vers la vue index
 - Dans la view `_company-selection.blade.php`, créez la sélection avec la liste **companies**



- Dans la view `_filter.blade.php`, écrire le code pour intégrer la sélection des companies

et la zone de recherche. (utiliser **include** ou **includeif** pour vérifier si la sous-vues existe)

- Utiliser `@includeUnless(empty($companies), 'contacts._company-selection')` qui nous permet d'afficher la sélection des companies en cas de la liste n'est pas vide .

6- En cas de la liste des contacte est vide, afficher le message : « aucun contact trouvé »

7- Utiliser `@each` pour afficher la liste des contacts :

`@each('contacts._contact', $contacts, 'contact', 'contacts._empty')`

`_empty.blade.php` est une view contient le message : Aucun contact trouvé

8- Réaliser la page d'accueil suivante :

