

Créer une application avec Laravel 5.5 – Les vues de l'authentification

Dans ce chapitre nous allons poursuivre le développement de l'application de galerie photos. On l'a laissée en chantier avec un layout adapté mais il faut maintenant qu'on s'occupe de l'apparence des formulaires de l'authentification.

Comme on va avoir des répétitions de code on va un peu mutualiser ça avec ce que nous offre Blade comme possibilités, en l'occurrence les inclusions de vues et les composants.

Enfin on va créer un helper pour rendre actif l'item en cours du menu dans la barre de navigation.

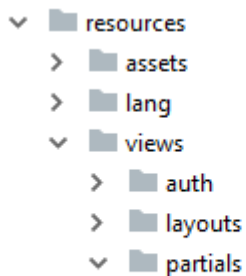
Les formulaires de Bootstrap 4

Si on regarde la documentation de Bootstrap 4 on a [une page consacrée aux formulaires](#). On voit que chaque contrôle est équipé de ce genre de code :

```
<div class="form-group">
  <label for="exampleInputEmail1">Email address</label>
  <input type="email" class="form-control" id="exampleInputEmail1"
  aria-describedby="emailHelp" placeholder="Enter email">
  <small id="emailHelp" class="form-text text-muted">We'll never
  share your email with anyone else.</small>
</div>
```

On a aussi [tout un passage concernant la validation](#).

Plutôt que de le reproduire plusieurs fois le même code on va créer une vue partielle à inclure :



form-group.blade.php
home.blade.php

En prévoyant toutes les variables nécessaires avec la validation :

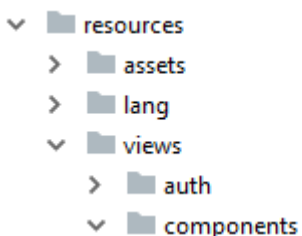
```
<div class="form-group">
  <label for="{{ $name }}">{{ $title }}</label>
  <input id="{{ $name }}" type="{{ $type }}" class="form-
control{{ $errors->has($name) ? ' is-invalid' : '' }}" name="{{
$name }}" value="{{ old($name, isset($value) ? $value : '') }}" {{
$required ? 'required' : ''}}>

  @if ($errors->has($name))
    <div class="invalid-feedback">
      {{ $errors->first($name) }}
    </div>
  @endif
</div>
```

Dans tous les formulaires on va aussi avoir un bouton de soumission. La syntaxe que nous donne la documentation est celle-ci :

```
<button type="submit" class="btn btn-primary">Submit</button>
```

Là on va créer un composant :



button.blade.php
layouts
partials
form-group.blade.php
home.blade.php

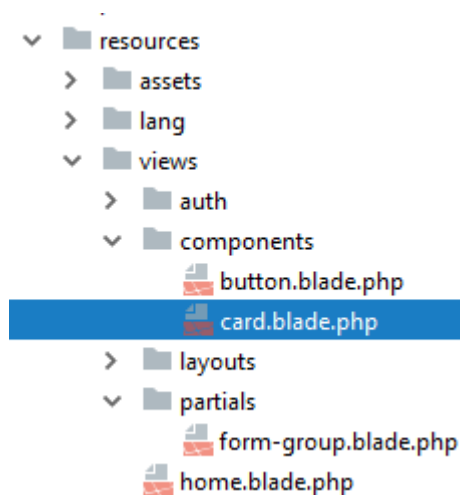
Avec ce code :

```
<button type="submit" class="btn btn-primary float-right">
    {{ $slot }}
</button>
```

Enfin pour faire plus joli on va inclure les formulaires dans [un composant card de Bootstrap](#). On voit dans la documentation la syntaxe de ce composant :

```
<div class="card">
  <div class="card-header">
    Featured
  </div>
  <div class="card-body">
    <h4 class="card-title">Special title treatment</h4>
    <p class="card-text">With supporting text below as a natural
lead-in to additional content.</p>
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
</div>
```

On va en faire un composant pour notre application :

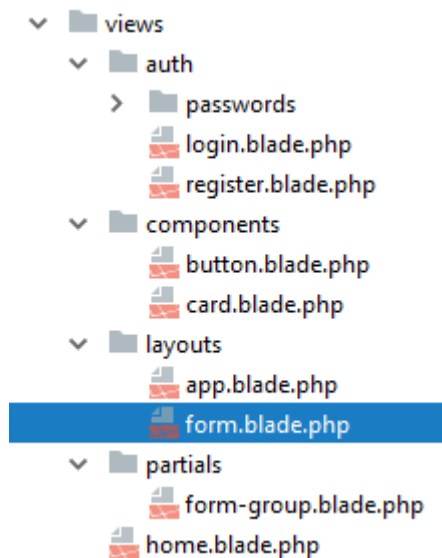


Avec ce code :

```
<div class="card text-white bg-dark mb-3">
  <h4 class="card-header">
    {{ $title }}
  </h4>
  <div class="card-body">
    {{ $slot }}
  </div>
```

```
</div>
```

Pour compléter la panoplie on va aussi créer un layout pour tous les formulaires :



```
@extends('layouts.app')
```

```
@section('content')
    <div class="container py-5">
        <div class="row">
            <div class="col-md-6 offset-md-3">
                @yield('card')
            </div>
        </div>
    </div>
@endsection
```

Maintenant on est bien équipé pour modifier les formulaires existant..

La connexion

Voici le nouveau code pour la vue de connexion (**auth/login**) :

```
@extends('layouts.form')

@section('card')

    @component('components.card')
```

```

@slot('title')
    @lang('Connexion')
@endslot

<form method="POST" action="{{ route('login') }}">
    {{ csrf_field() }}

    @include('partials.form-group', [
        'title' => __('Adresse email'),
        'type' => 'email',
        'name' => 'email',
        'required' => true,
    ])

    @include('partials.form-group', [
        'title' => __('Mot de passe'),
        'type' => 'password',
        'name' => 'password',
        'required' => true,
    ])

    <div class="custom-control custom-checkbox">
        <input type="checkbox" class="custom-control-
input" id="remember" name="remember" {{ old('remember') ?
'checked' : '' }}>
            <label class="custom-control-label"
for="remember"> @lang('Se rappeler de moi')</label>
        </div>

    @component('components.button')
        @lang('Connexion')
    @endcomponent

        <a class="btn btn-link" href="{{
route('password.request') }}">
            @lang('Mot de passe oublié ?')
        </a>

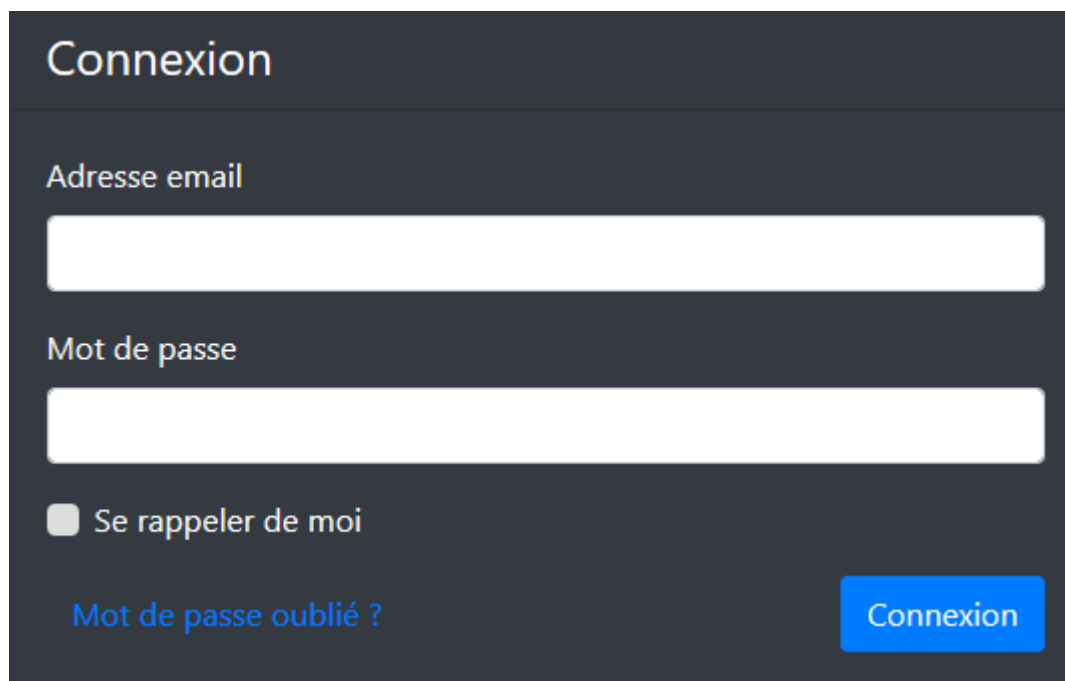
    </form>

@endcomponent

@endsection

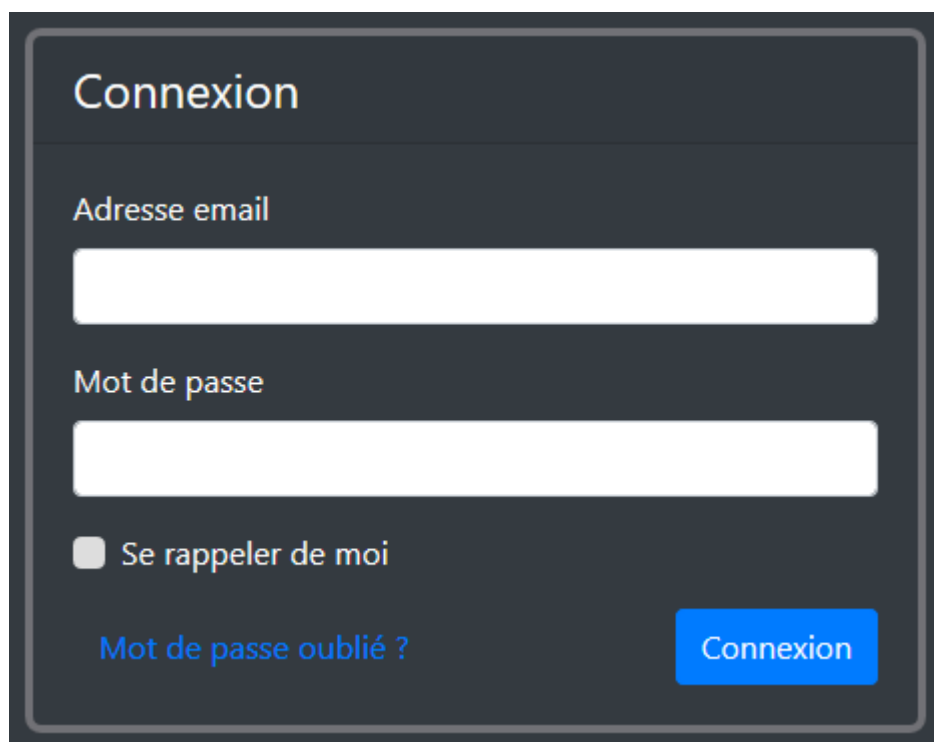
```

Ce qui donne cet aspect :



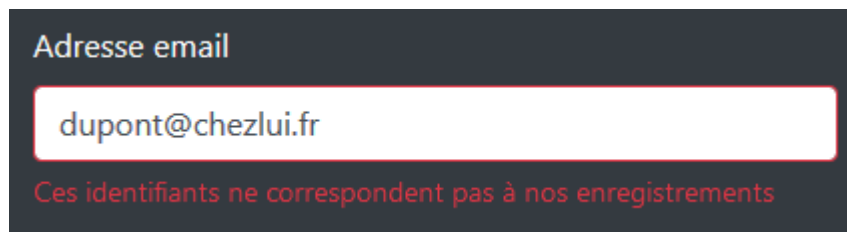
Pour bien délimiter le formulaire on va dessiner la bordure avec le CSS :

```
.card {  
  border: 4px solid rgba(255, 242, 242, .3);  
  border-radius: .5rem;  
}
```



*Pensez à activer les modifications avec **npm***

On vérifie que la validation apparaît bien :



A screenshot of a dark-themed web form. At the top, the text "Adresse email" is displayed. Below it is a white input field containing the email address "dupont@chezlui.fr". A red border highlights the input field. Below the input field, a red error message reads "Ces identifiants ne correspondent pas à nos enregistrements".

L'inscription

De la même manière on va construire la vue pour l'inscription (**auth/register**) :

```
@extends('layouts.form')
```

```
@section('card')
```

```
    @component('components.card')
```

```
        @slot('title')
```

```
            @lang('Inscription')
```

```
        @endslot
```

```
        <form method="POST" action="{{ route('register') }}">
            {{ csrf_field() }}
```

```
            @include('partials.form-group', [
                'title' => __('Nom'),
                'type' => 'text',
                'name' => 'name',
                'required' => true,
            ])
```

```
            @include('partials.form-group', [
                'title' => __('Adresse email'),
                'type' => 'email',
                'name' => 'email',
                'required' => true,
            ])
```

```
            @include('partials.form-group', [
                'title' => __('Mot de passe'),
```

```
'type' => 'password',  
'name' => 'password',  
'required' => true,  
)
```

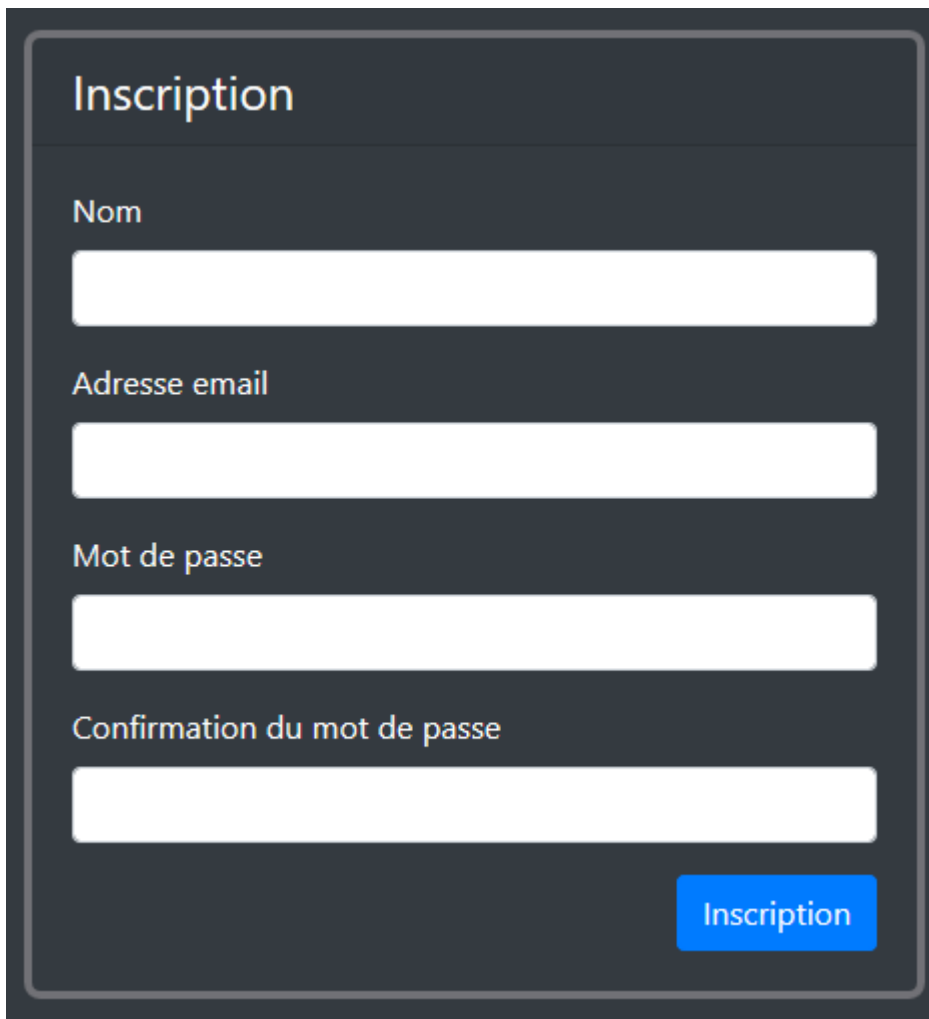
```
@include('partials.form-group', [  
  'title' => __('Confirmation du mot de passe'),  
  'type' => 'password',  
  'name' => 'password_confirmation',  
  'required' => true,  
)
```

```
@component('components.button')  
  @lang('Inscription')  
@endcomponent
```

```
</form>
```

```
@endcomponent
```

```
@endsection
```



The image shows a dark-themed registration form titled "Inscription". It contains four input fields: "Nom", "Adresse email", "Mot de passe", and "Confirmation du mot de passe". A blue button labeled "Inscription" is located at the bottom right of the form.

Le renouvellement du mot de passe

On continue avec la vue pour la demande de renouvellement du mot de passe (**auth/email**) :

```
@extends('layouts.form')

@section('card')

    @if (session('status'))
        <div class="alert alert-success" role="alert">
            {{ session('status') }}
        </div>
    @endif

    @component('components.card')

        @slot('title')
            @lang('Renouvellement du mot de passe')
        @endslot

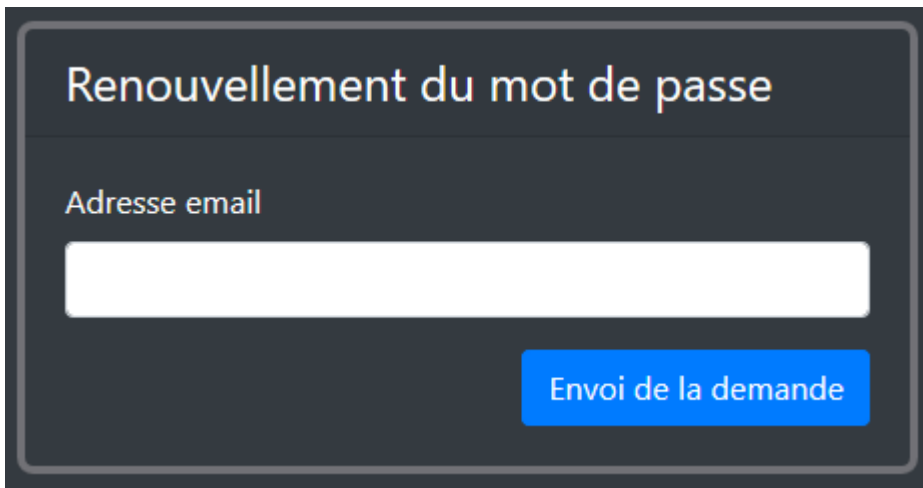
        <form method="POST" action="{{ route('password.email')
    }}">
            {{ csrf_field() }}

            @include('partials.form-group', [
                'title' => __('Adresse email'),
                'type' => 'email',
                'name' => 'email',
                'required' => true,
            ])

            @component('components.button')
                @lang('Envoi de la demande')
            @endcomponent
        </form>

    @endcomponent

@endsection
```



Et enfin la vue pour le renouvellement (**auth/reset**) :

```
@extends('layouts.form')
```

```
@section('card')
```

```
    @component('components.card')
```

```
        @slot('title')
```

```
            @lang('Renouvellement du mot de passe')
```

```
        @endslot
```

```
        <form method="POST" action="{{ route('password.request') }}">
```

```
            {{ csrf_field() }}
```

```
            <input type="hidden" name="token" value="{{ $token }}">
```

```
            @include('partials.form-group', [
                'title' => __('Adresse email'),
                'type' => 'email',
                'name' => 'email',
                'required' => true,
            ])
```

```
            @include('partials.form-group', [
                'title' => __('Mot de passe'),
                'type' => 'password',
                'name' => 'password',
                'required' => true,
            ])
```

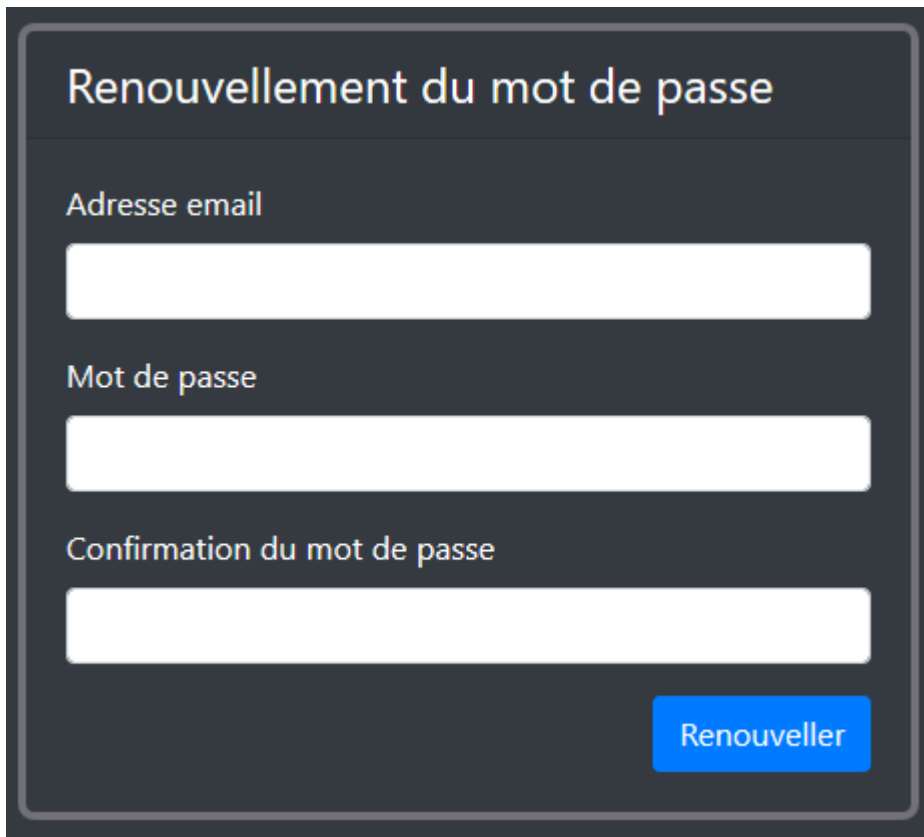
```
@include('partials.form-group', [  
    'title' => __('Confirmation du mot de passe'),  
    'type' => 'password',  
    'name' => 'password_confirmation',  
    'required' => true,  
])
```

```
@component('components.button')  
    @lang('Renouveler')  
@endcomponent
```

```
</form>
```

```
@endcomponent
```

```
@endsection
```



The image shows a dark-themed form titled "Renouvellement du mot de passe". It contains three input fields: "Adresse email", "Mot de passe", and "Confirmation du mot de passe". A blue button labeled "Renouveler" is positioned at the bottom right of the form.

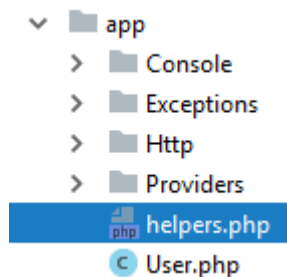
L'item actif dans le menu

On a bien avancé mais il y a une chose pas très jolie au niveau de la barre de navigation : rien n'indique quel item est actif. On va maintenant arranger ça...

Quand on regarde [la documentation de Bootstrap 4](#) on voit qu'il suffit d'ajouter la classe **active** à un item pour le distinguer visuellement :

```
<li class="nav-item active">
```

Il faut donc qu'on s'arrange pour ajouter cette classe en fonction de la requête en cours. Pour l'occasion on va créer un helper. On commence par créer un fichier (**app/helpers.php**) :



Et on lui ajoute cette fonction :

```
<?php
```

```
if (!function_exists('currentRoute')) {
    function currentRoute(...$routes)
    {
        foreach($routes as $route) {
            if(request()->url() == $route) {
                return ' active';
            }
        }
    }
}
```

On prend toujours la précaution de vérifier que la fonction n'existe pas déjà. Pour que Laravel soit au courant que notre fonction existe on va ajouter la référence dans **composer.json** :

```
"autoload": {
    "classmap": [
        "database/seeds",
        "database/factories"
    ],
    "files": [
        "app/helpers.php"
    ],
}
```

```
    "psr-4": {
        "App\\": "app/"
    }
},
```

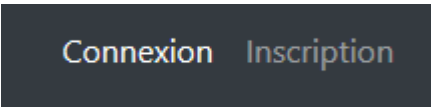
On relance composer :

```
composer dumpautoload
```

Maintenant il ne reste plus qu'à modifier le code de la barre de navigation dans la vue **layouts/app** :

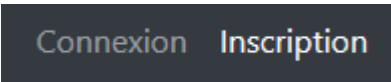
```
<li class="nav-item{{ currentRoute(route('login')) }}"><a
class="nav-link" href="{{ route('login')
}}">@lang('Connexion')</a></li>
<li class="nav-item{{ currentRoute(route('register')) }}"><a
class="nav-link" href="{{ route('register')
}}">@lang('Inscription')</a></li>
```

Maintenant quand je vais sur la vue de connexion l'item correspondant s'illumine :



Connexion Inscription

Et ça fonctionne aussi pour l'inscription :



Connexion Inscription

Il n'y a que pour le renouvellement du mot de passe qu'on a rien à rendre actif...

Conclusion

Dans ce chapitre on a vu :

- comment créer une vue à inclure et des composants pour faciliter l'écriture des vue de formulaire
- comme styliser les formulaires
- comment créer un helper pour rendre actif l'item en cours dans la barre de navigation.

Pour vous simplifier la vie vous pouvez [charger le projet](#) dans son état à l'issue de ce chapitre.