

# Laravel Pour débiter

## I. Installer Laravel

1. `composer selfupdate`
2. déplacez- vous pour aller dans le répertoire de publication (www sous Windows)
3. Ouvrez une console dans ce répertoire
4. Installer l'application  
`composer create-project --prefer-dist laravel/laravel gsbLaravel`
5. Copier dans le répertoire gsbLaravel du répertoire de publication (www sous windows) l'ensemble des fichiers dézippés ; vous répondrez que vous voulez modifier à chaque fois que la question est posée.
6. Il faudra modifier si nécessaire les paramètres de connexion du fichier .env

## II. Bug à l'utilisation

Une fois installé, avec laragon, vous pouvez rencontrer des erreurs avec les routes (erreur 404).

Dans ce cas il faut lancer le serveur interne de laravel.

1. Lancer le terminal
2. se déplacer dans le répertoire ou se trouve gsbLaravel
3. saisir `php artisan serve`
4. Dans le navigateur saisir <http://127.0.0.1:8000>

## III. Debugger votre programme

1. L'instruction **dd()** affiche les données passées en paramètre et stoppe le programme
2. L'instruction **dump()** affiche les données passées en paramètre et continue l'exécution
3. Les log sont visibles dans le répertoire **storage/logs**
4. Si la page d'erreur n'apparaît pas vous pouvez installer telescope  
`php artisan telescope:install`

## IV. La procédure à suivre

### A. Le menu

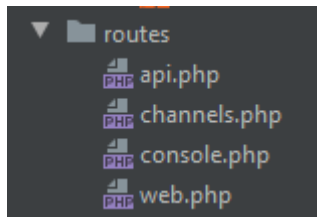
Sommaire.blade.php

Il permet d'indiquer à Laravel les routes à prendre pour effectuer des opérations.

```
<li class="smenu">
  <a href="{{ route('chemin_gestionFrais') }}" title="Saisie fiche de frais" >Saisie fiche de frais</a>
</li>
```

Ici en cliquant sur **saisie frais** on appelle la route **chemin\_gestionFrais**

## B. Les routes



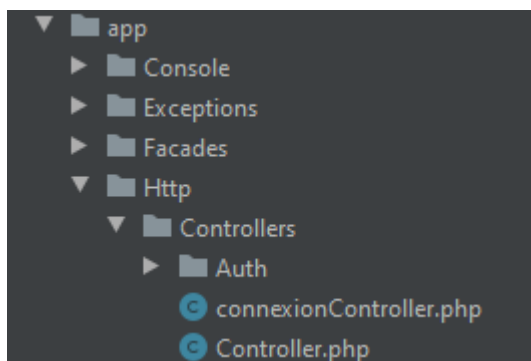
Elles se trouvent dans le fichier web.php

```
Route::get('gererFrais', [
    'as' => 'chemin_gestionFrais',
    'uses' => 'gererFraisController@saisirFrais'
]);
```

- gererFrais : le mot qui apparait dans l'url
- gererFraisController@saisirFrais : appel sous la forme contrôleur @action

Ici la route va chercher dans le contrôleur **gererFraisController** la fonction **saisirFrais**

## C. Les contrôleurs



Voici le contenu du fichier :

Ici se trouve la définition du contrôleur

```
namespace App\Http\Controllers;
use Illuminate\Http\Request;

use PdoGsb;
use MyDate;
class gererFraisController extends Controller{

    //Ici se trouve l'action liée au contrôleur
    function saisirFrais(Request $request){
        if( session('visiteur') != null){ // Test de la session
            $visiteur = session('visiteur'); // Capture de la session
            $idVisiteur = $visiteur['id']; // Capture du visiteur
            $anneeMois = MyDate::getAnneeMoisCourant();
            $mois = $anneeMois['mois'];
            if(PdoGsb::estPremierFraisMois($idVisiteur,$mois)){
```

```

        PdoGsb::creeNouvellesLignesFrais($idVisiteur,$mois); // Appel du modèle
    }

    $lesFrais = PdoGsb::getLesFraisForfait($idVisiteur,$mois);
    $view = view('majFraisForfait') // vue à afficher
        ->with('lesFrais', $lesFrais) // paramètres de la vue
        ->with('numMois',$anneeMois['numMois'])
        ->with('erreurs',null)
        ->with('numAnnee',$anneeMois['numAnnee'])
        ->with('visiteur',$visiteur)
        ->with('message',"")
        ->with('method',$request->method());
    return $view; // Affichage de la vue
}

else{ // Vue à afficher sans session
    return view('connexion')->with('erreurs',null);
}
}
}

```

## D. Les vues

Elles sont déclarées de la façon suivante :

nomVue.blade.php

A part la vue connexion et la vue sommaire, les vues sont faites de cette façon :

```
@extends('sommaire') //Equivalent de l'include
@section('contenu1') //Le contenu à ajouter

<div id="contenu">
    <h2>Renseigner ma fiche de frais du mois {{ $numMois }}-{{ $numAnnee }}</h2>
    <form method="post" action="{{ route('chemin_sauvegardeFrais') }}">
        {{ csrf_field() }} <!-- laravel va ajouter un champ caché avec
un token -->
        <div class="corpsForm">
            <fieldset>
                <legend>Eléments forfaitisés</legend>
                @includeWhen($erreurs != null, 'msgerreurs', ['erreurs' =>
$erreurs])
                @includeWhen($message != "", 'message', ['message' => $message])
                @foreach ($lesFrais as $key => $frais)
                    <p>
                        <input type = "hidden" name = "lesLibFrais[]"
                            @if($method == 'GET')
                                value = "{{ $frais['libelle'] }}"
                            @else
                                value = "{{ $lesLibFrais[$loop->index] }}"
                            @endif>
                        <label name = "libelle" for="idFrais">
                            @if($method == 'GET')
                                {{ $frais['libelle'] }} //Insertion de code php
                            @else
                                {{ $lesLibFrais[$loop->index] }} //sans point-virgule
                            @endif
                        </label>
                        <input type="text" required
```

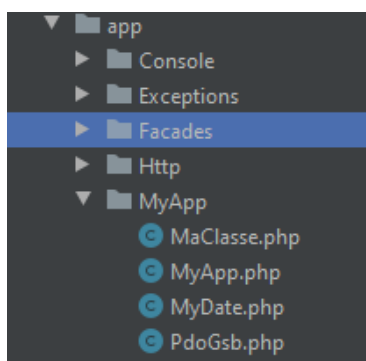
```

        @if($method == 'GET')
            name = "lesFrais[{{$frais['idfrais']}}]"
            value = "{{$frais['quantite']}}"
        @else
            name = "lesFrais[{{$key}}]"
            value = "{{$frais}}"
        @endif
    </p>
    @endforeach
</fieldset>
</div>
<div class="piedForm">
    <p>
        <input id="ok" type="submit" value="Valider" size="20" />
        <input id="annuler" type="reset" value="Annuler" size="20" />
    </p>
</div>
</form>
</div>
@endsection //Fin de contenu

```

## E. Le modèle

Vous savez l'utiliser



## V. Exemple : créer une page test dans EtatFrais

Cet Exemple est valable pour le visiteur, à vous de l'adapter pour un autre type d'intervenant.

### F. Le sommaire

```

<li class="smenu">
    <a href="{{ route('chemin_test') }}" title="test">test</a>
</li>

```

### G. La route

```

Route::get('test',[
    'as'=>'chemin_test',
    'uses'=>'etatFraisController@test'
]);

```

### H. Le contrôleur

```

function test(){
    if( session('visiteur')!= null){ //Sans la session l'insertion du sommaire
        $visiteur = session('visiteur'); //provoque une erreur
        $idVisiteur = $visiteur['id'];
        return view('test') ->with('visiteur',$visiteur);
    }
}

```

```
}  
else{  
    return view('connexion')->with('erreurs',null);  
}  
}
```

#### I. La vue

```
@extends ('sommaire')  
@section('contenu1')  
    <h1>titre</h1>  
@endsection
```

## VI. Les erreurs

### There is no existing directory at ...

1. Lancer le terminal
2. Aller dans le repertoire de laravel
3. Saisir php artisan optimize:clear