

Step 1: Laravel 10 PROJECT

`composer create-project laravel/laravel dual-language-test`

`cd dual-language-test`

Step 2: Create Lang Files

By default, the Laravel 10 application skeleton does not include the lang directory.

If you would like to customize Laravel's language files, you may publish them via the `lang:publish` Artisan command.

`php artisan lang:publish`

step 3:

in lang folder, make another folder named ur (for urdu)
make a file `messages.php` in both en and ur folders

step 4:

write the below in `en/messages` file

`<?php`

```
return [  
    'save' => 'Save',  
    'update' => 'Update',  
    'ok' => 'Ok',  
];
```

step 5:

translate these messages and write in `ur/messages.php` file

step 6: Create Routes

write the below line in imports section of `web.php`

`use App\Http\Controllers\LangController;`

step 7:

make below 2 routes in `web.php`

`Route::get('lang/home', [LangController::class, 'index']);`

`Route::get('lang/change', [LangController::class, 'change'])->name('changeLang');`

step 8:

make a controller by the name of `LangController`

`php artisan make:controller LangController`

step 9:

make the following two functions in it

```
public function index()

{

    return view('lang');

}


/**

 * Display a listing of the resource.

 *

 * @return \Illuminate\Http\Response

 */

public function change(Request $request)

{

    App::setLocale($request->lang);

    session()->put('locale', $request->lang);


    return redirect()->back();

}
```

Step 10: Create View

make lang.blade.php in resources views folder

step 11:

paste the below code inside it

```
<!DOCTYPE html>
<html>

<head>
    <title>Multi Language Website</title>
```

```

<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="//netdna.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
</head>

<body>
<div class="container">
    <h1>How to Create Multi Language Website in Laravel - ItSolutionStuff.com</h1>

    <div class="row">
        <div class="col-md-2 col-md-offset-6 text-right">
            <strong>Select Language: </strong>
        </div>

        <div class="col-md-4">
            <select class="form-control changeLang">
                <option value="en" {{ session()->get('locale') == 'en' ? 'selected' : "
}}}>English</option>
                <option value="fr" {{ session()->get('locale') == 'ur' ? 'selected' : "
}}}>Urdu</option>
            </select>
        </div>

    </div>

    <h1>{{ __('messages.save') }}</h1>

</div>
</body>

<script type="text/javascript">
    var url = "{{ route('changeLang') }}";

    $(".changeLang").change(function() {
        window.location.href = url + "?lang="+ $(this).val();
    });

</script>

</html>

```

step 12: Create Middleware

php artisan make:middleware LanguageManager

step 13:

Now you have to update middleware file like bellow:

app/Http/Middleware/LanguageManager.php

```
<?php
```

```
namespace App\Http\Middleware;
```

```
use Closure;
```

```
use App;
```

```
class LanguageManager
```

```
{
```

```
    /**
```

```
     * Handle an incoming request.
```

```
     *
```

```
     * @param \Illuminate\Http\Request $request
```

```
     * @param \Closure $next
```

```
     * @return mixed
```

```
     */
```

```
    public function handle($request, Closure $next)
```

```
    {
```

```
        if (session()->has('locale')) {
```

```
            App::setLocale(session()->get('locale'));
        }
```

```
    }
```

```
        return $next($request);
```

```
    }
```

```
}
```

step 14:

Now we need to register the newly created middleware to kernel file.

so let's add it as bellow:

app/Http/Kernel.php

in the \$middlewareGroups section

add the below line

```
\App\Http\Middleware\LanguageManager::class,
```

step 15:

Run Laravel App: