



Laravel Visitor

This is a laravel package to extract and access visitors' information such as `browser` , `ip` , `device` and etc.

In this package, you can recognize online users and determine if a user is online or not

Install

Via composer

```
composer require shetabit/visitor
```

Configure

If you are using Laravel 5.5 or higher then you don't need to add the provider and alias.

```
# In your providers array.
'providers' => [
    ...
    Shetabit\Visitor\Provider\VisitorServiceProvider::class,
],

# In your aliases array.
'aliases' => [
    ...
    'Visitor' => Shetabit\Visitor\Facade\Visitor::class,
],
```

Then, run the below commands to publish migrations and create tables

```
php artisan vendor:publish

php artisan migrate
```

How to use

You can access to `visitor's` information using `$request->visitor()` in your controllers , and you can access to the visitor's information using `visitor()` helper function any where.

We have the below methods to retrieve a visitor's information:

- `device` : device's name
- `platform` : platform's name
- `browser` : browser's name
- `languages` : language's name
- `ip` : client's ip
- `request` : the whole request inputs
- `useragent` : the whole useragent
- `isOnline` : determines if current (or given) user is online

```
$request->visitor()->browser(); // firefox
$request->visitor()->visit($post); // create log for post
$request->visitor()->setVisitor($user)->visit($post); // create a log wich says $user has visited $post
```

Store Logs

You can create logs using the `visit` method like the below

```
visitor()->visit(); // create a visit log
```

use `Shetabit\Visitor\Traits\Visitable` trait in your models, then you can save visit's log for your models like the below

```
// or you can save log like the below
visitor()->visit($model);
// or like the below
$model->createVisitLog();

// you can say which user has visited the given $model
$model->createVisitLog($user);
// or like the below
visitor()->setVisitor($user)->visit($model);
```

Model views can be loaded using `visits` relation.

You can count model visits like the below

```
$model->visitLogs()->count();
```

unique users can be counted by their IP and by model.

```
// by ip
$model->visitLogs()->distinct('ip')->count('ip');

// by user's model
$model->visitLogs()->visitor()->count();
```

use `Shetabit\Visitor\Traits\Visitor` in your `User` class, then you can run below codes

```
$user->visit(); // create a visit log
$user->visit($model); // create a log which says, $user has visited $model
```

Retrieve and Determine Online users

use `Shetabit\Visitor\Traits\Visitor` in your `User` class at first.

Then you can retrieve online users which are instance of `User` class and determine if a user is online.

```
visitor()->onlineVisitors(User::class); // returns collection of online users
User::online()->get(); // another way

visitor()->isOnline($user); // determines if the given user is online
$user->isOnline(); // another way
```

Automatic logging

Your application can store visitor's log automatically using `LogVisits` middleware.

Add the `Shetabit\Visitor\Middlewares\LogVisits` middleware if you want to save logs automatically.

The middleware will store logs for models which has binded in router (router model binding) and has used `Shetabit\Visitor\Traits\Visitable` trait.