APPENDIX C

(Source Code)

**Web Controllers (PHP Laravel):**

**HomeController.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class HomeController extends Controller

{

public function index(){

return view('home.home');

}

}

**FindController.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Support\Facades\DB;

use Illuminate\Http\Request;

class FindController extends Controller

{

public function index()

{

$train\_count = DB::table('TrainInfo')->count();

if($train\_count!=0){

return view('find.find');

}

else{

return view('find.train');

}

}

public function upload(Request $request)

{

$file = $request->file('image');

$ext = array('jpg','JPEG','bmp','png', 'JPG');

$valid = false;

foreach($ext as $extension){

if($file->getClientOriginalExtension() == $extension){

$valid = true;

}

}

if(!$valid)

{

return view('find.find')->with('valid', 'false');

}

else{

# $storage = base\_path('eigencore/test/recognize');

$storage = base\_path('public/img/faces/find');

$image\_name = $this->GUIDRANDOM();

$file\_name = $image\_name.".".$file->getClientOriginalExtension();

$file->move($storage,$file\_name);

$python\_script = base\_path('eigen-core/src/main.py demo');

$image\_to\_recog = $storage.'/'.$file\_name;

$exec\_query = '/usr/bin/python '.$python\_script.' '.$image\_name.' '.$image\_to\_recog;

exec($exec\_query , $output);

$path\_like = $output[2].'%';

$result = DB::table('TrainInfo')->where('file', 'like', $path\_like)->take(1)->get();

// var\_dump($exec\_query);

// var\_dump($output);

// # static path for display

$storage\_path\_1 = 'img/faces/find/';

$storage\_path\_2 = 'img/faces/found/';

$imagepath = $storage\_path\_1.$file\_name;

$matchpath = $storage\_path\_2.$result[0]->file;

$runtime = $output[1];

return view('find.train')

->with('uploaded', 'true')

->with('result', $result)

->with('matchpath', $matchpath)

->with('imagepath', $imagepath)

->with('runtime', $runtime);

}

}

private function GUIDRANDOM ($trim = true)

{

// Windows

if (function\_exists('com\_create\_guid') === true) {

if ($trim === true)

return trim(com\_create\_guid(), '{}');

else

return com\_create\_guid();

}

// OSX/Linux

if (function\_exists('openssl\_random\_pseudo\_bytes') === true) {

$data = openssl\_random\_pseudo\_bytes(16);

$data[6] = chr(ord($data[6]) & 0x0f | 0x40); // set version to 0100

$data[8] = chr(ord($data[8]) & 0x3f | 0x80); // set bits 6-7 to 10

return vsprintf('%s%s-%s-%s-%s-%s%s%s', str\_split(bin2hex($data), 4));

}

// Fallback (PHP 4.2+)

mt\_srand((double)microtime() \* 10000);

$charid = strtolower(md5(uniqid(rand(), true)));

$hyphen = chr(45); // "-"

$lbrace = $trim ? "" : chr(123); // "{"

$rbrace = $trim ? "" : chr(125); // "}"

$guidv4 = $lbrace.

substr($charid, 0, 8).$hyphen.

substr($charid, 8, 4).$hyphen.

substr($charid, 12, 4).$hyphen.

substr($charid, 16, 4).$hyphen.

substr($charid, 20, 12).

$rbrace;

return $guidv4;

}

}

**FoundController.php**

<?php

namespace App\Http\Controllers;

use Illuminate\Support\Facades\DB;

use Illuminate\Http\Request;

use Carbon\Carbon;

class FoundController extends Controller

{

public function index(){

//echo "<h1>Found<h1>";

$result = false;

return view('found.found')->with('result', $result);

}

public function upload(Request $request)

{

$name = $request->input('name');

$message = $request->input('message');

$location = $request->input('location');

$file = $request->file('image');

$ext = array('jpg','JPEG','bmp','png', 'JPG');

$valid = false;

$result = false;

$oldImg = "";

$newImg = "";

foreach($ext as $extension){

if($file->getClientOriginalExtension() == $extension){

$valid = true;

}

}

if(!$valid)

{

return view('found.found')

->with('valid', 'false')

->with('result', $result);

}

else{

# $storage = base\_path('eigencore/test/teach');

$storage = base\_path('public/img/faces/found');

$image\_name = $this->GUIDRANDOM();

$file\_name = $image\_name.".".$file->getClientOriginalExtension();

$file->move($storage,$file\_name);

$now = date('Y-m-d H:i:s');

$teach = base\_path('eigen-core/src/main.py train');

$exec\_query = '/usr/bin/python '.$teach.' '.$image\_name.' '.$storage.'/'.$file\_name;

exec($exec\_query , $output);

if($output[0] == "Image Trained Successfully!"){

DB::table('TrainInfo')->insert(

['name' => $name,

'location' => $location,

'message' => $message,

'found' => false,

'file' => $file\_name,

'created\_at' => $now,

'updated\_at' => $now,

'state' => $output[1],

'old\_mean' => $output[2],

'new\_mean' => $output[3],

]);

$result = true;

$oldImg = 'img/faces/training\_images\_old/'.$image\_name.'/1.jpg';

$newImg = 'img/faces/training\_images/'.$image\_name.'/1.jpg';

return view('found.found')

->with('uploaded', 'true')

->with('result', $result)

->with('old\_image', $oldImg)

->with('new\_image', $newImg);

}

else{

return view('found.found')

->with('uploaded', 'false')

->with('result', $result);

}

}

}

function GUIDRANDOM ($trim = true)

{

// Windows

if (function\_exists('com\_create\_guid') === true) {

if ($trim === true)

return trim(com\_create\_guid(), '{}');

else

return com\_create\_guid();

}

// OSX/Linux

if (function\_exists('openssl\_random\_pseudo\_bytes') === true) {

$data = openssl\_random\_pseudo\_bytes(16);

$data[6] = chr(ord($data[6]) & 0x0f | 0x40); // set version to 0100

$data[8] = chr(ord($data[8]) & 0x3f | 0x80); // set bits 6-7 to 10

return vsprintf('%s%s-%s-%s-%s-%s%s%s', str\_split(bin2hex($data), 4));

}

// Fallback (PHP 4.2+)

mt\_srand((double)microtime() \* 10000);

$charid = strtolower(md5(uniqid(rand(), true)));

$hyphen = chr(45); // "-"

$lbrace = $trim ? "" : chr(123); // "{"

$rbrace = $trim ? "" : chr(125); // "}"

$guidv4 = $lbrace.

substr($charid, 0, 8).$hyphen.

substr($charid, 8, 4).$hyphen.

substr($charid, 12, 4).$hyphen.

substr($charid, 16, 4).$hyphen.

substr($charid, 20, 12).

$rbrace;

return $guidv4;

}

}

**Routes:  
web.php**

<?php

Route::get('/', 'HomeController@index');

Route::get('home/', 'HomeController@index');

Route::get('found/', 'FoundController@index');

Route::get('find/', 'FindController@index');

Route::post('found/', 'FoundController@upload');

Route::post('find/', 'FindController@upload');

**Eigenface Core (Python)**

**main.py**

# -\*- coding: utf-8 -\*-

import eigenfaces

ef = eigenfaces.EigenFaces()

ef.train("/home/aethelbert/Programming/PROGRAMMING/trackery-core-obj-1/public/img/faces/training\_images")

ef.show\_results()

predicted\_class = ef.predict\_face\_in\_image(0)

print(predicted\_class)

eigenfaces.show\_image\_for(predicted\_class)

**eigenface.py**

# -\*- coding: utf-8 -\*-

from PIL import Image

import numpy as np

import pylab

import sys

import glob

import os

import pca

class EigenFaces(object):

def train(self, root\_training\_images\_folder):

self.projected\_classes = []

self.list\_of\_arrays\_of\_images, self.labels\_list, \

list\_of\_matrices\_of\_flattened\_class\_samples = \

read\_images(root\_training\_images\_folder)

# create matrix to store all flattened images

images\_matrix = np.array([np.array(Image.fromarray(img)).flatten()

for img in self.list\_of\_arrays\_of\_images],'f')

# perform PCA

self.eigenfaces\_matrix, variance, self.mean\_Image = pca.pca(images\_matrix)

# Projecting each class sample (as class matrix) and then using the class average as the class weights for comparison with the Target image

for class\_sample in list\_of\_matrices\_of\_flattened\_class\_samples:

class\_weights\_vertex = self.project\_image(class\_sample)

self.projected\_classes.append(class\_weights\_vertex.mean(0))

def predict\_face\_in\_image(self, image\_nr):

target\_images = self.get\_target\_images()

ti = np.array(Image.open(target\_images[0]), dtype = np.uint8).flatten()

return self.predict\_face(ti)

def show\_results(self):

anImage = np.array(Image.fromarray(self.list\_of\_arrays\_of\_images[0]))

image\_height, image\_width = anImage.shape[0:2] # get the size of the images

pylab.figure()

pylab.gray()

pylab.subplot(2, 4, 1)

pylab.imshow(self.mean\_Image.reshape(image\_height, image\_width))

for i in range(7):

pylab.subplot(2, 4, i+2)

pylab.imshow(self.eigenfaces\_matrix[i].reshape(

image\_height, image\_width))

def extract(self,X):

X = np.asarray(X).reshape(-1, 1)

return self.project\_image(X)

def project\_image(self, X):

X = X - self.mean\_Image

return np.dot(X, self.eigenfaces\_matrix.T)

def reconstruct(self, X):

X = np.dot(X, self.eigenfaces\_matrix)

return X + self.mean\_Image

def get\_target\_images(self):

return glob.glob('target\_image/\*.jpg')

def predict\_face(self, X):

min\_class = -1

min\_distance = np.finfo('float').max

projected\_target = self.project\_image(X)

# delete last array item, it's nan

projected\_target = np.delete(projected\_target, -1)

for i in range(len(self.projected\_classes)):

distance = np.linalg.norm(projected\_target - np.delete(self.projected\_classes[i], -1))

if distance < min\_distance:

# print distance, min\_distance, True

min\_distance = distance

min\_class = self.labels\_list[i]

else:

if (distance - min\_distance) < 800:

# print distance, min\_distance, False

min\_class = "Unknown."

else:

min\_class = self.labels\_list[i]

return min\_class

def predict\_race(self, X):

return np.min\_target

def get\_class\_average\_from\_samples(class\_samples):

m, n = np.array(class\_samples).shape[1:3]

l = len(class\_samples)

add\_samples\_together = np.zeros((m, n))

for a in class\_samples:

add\_samples\_together = np.add(add\_samples\_together, a)

averagedClass = np.divide(add\_samples\_together, l)

return averagedClass

def \_\_repr\_\_(self):

return "PCA (num\_components=%d)" % (self.\_num\_components)

def show\_image\_for(img\_class):

predicted\_img = "training\_images/%s/1.jpg" % (img\_class)

img = Image.open(predicted\_img)

img.show()

def read\_images(path, sz=None):

"""Reads the images in a given folder, resizes images on the fly if size is given.

Args:

path: Path to a folder with subfolders representing the subjects (persons).

sz: A tuple with the size Resizes

Returns:

A tuple of (images, image\_labels, class\_matrix\_list) where

images: The images, which is a Python list of numpy arrays.

image\_labels: The corresponding labels (the unique number of

the subject, person).

"""

class\_samples\_list = []

class\_matrices\_list = []

images, image\_labels = [], []

for dirname, dirnames, filenames in os.walk(path):

for subdirname in dirnames:

subject\_path = os.path.join(dirname, subdirname)

class\_samples\_list = []

for filename in os.listdir(subject\_path):

if filename != ".DS\_Store":

try:

im = Image.open(os.path.join(subject\_path, filename))

# resize to given size (if given) e.g., sz = (480, 640)

if (sz is not None):

im = im.resize(sz, Image.ANTIALIAS)

images.append(np.asarray(im, dtype = np.uint8))

except IOError as e:

errno, strerror = e.args

print("I/O error({0}): {1}".format(errno, strerror))

except:

print("Unexpected error:", sys.exc\_info()[0])

raise

# adds each sample within a class to this List

class\_samples\_list.append(np.asarray(im, dtype = np.uint8))

# flattens each sample within a class and adds the array/vector to a class matrix

class\_samples\_matrix = np.array([img.flatten()

for img in class\_samples\_list],'f')

# adds each class matrix to this MASTER List

class\_matrices\_list.append(class\_samples\_matrix)

image\_labels.append(subdirname)

return images, image\_labels, class\_matrices\_list

**pca.py**

import numpy as np

import timeit as t

def pca(X):

"""Principal Component Analysis

input: X, matrix with training data stored as flattened arrays in rows

return: projection matrix (with most important dimensions first).

"""

# get dimensions

num\_data, dim = X.shape

# center data

mean\_X = X.mean(axis=0)

X = X - mean\_X

start = t.default\_timer()

if dim > num\_data:

# PCA - compact trick used

print "PCA DIMENSION REDUCTION METHOD"

M = np.dot(X,X.T) # covariance matrix

e,EV = np.linalg.eigh(M) # eigenvalues and eigenvectors

tmp = np.dot(X.T,EV).T # this is the compact trick

V = tmp[::-1] # reverse since last eigenvectors are the ones we want

S = np.sqrt(e[::-1]) # reverse since eigenvalues are in increasing order

S = 50

for i in range(V.shape[1]):

V[:,i] /= S

stop = t.default\_timer()

end = stop - start

print "time: {}s".format(end)

else:

# PCA - SVD used

print "PCA SINGLE VALUE DECOMPOSITION"

U, S, V = np.linalg.svd(X)

V = V[:num\_data] # only makes sense to return the first num\_data

stop = t.default\_timer()

end = stop - start

print end

# return the projection matrix, the variance and the mean

return V, S, mean\_X

**gfx.py**

"""Graphics utilities."""

import cv2

import numpy as np

from eigenfaces.eigenfaces import EigenFaces

class Image:

"""Image manipulation class wrapping some opencv functions."""

def \_\_init\_\_(self, ocv\_image):

"""Wrap opencv image."""

self.\_img = ocv\_image

self.\_face\_cascade = "/home/aethelbert/Programming/PROGRAMMING/trackery-core-obj-1/eigen-core/haarcascade\_frontalface\_default.xml"

def gray(self):

"""

Returns:

new grayscale image.

"""

img = Image(cv2.cvtColor(self.\_img, cv2.COLOR\_BGR2GRAY))

return img

def faces(self):

return map(lambda rect: self.cut(rect[0], rect[1], rect[2], rect[3]),

self.face\_areas())

def cut(self, x, y, width, height):

return Image(self.\_img[y:y + height, x:x + width])

def scale(self, width, height):

return Image(cv2.resize(self.\_img, (width, height)))

def draw\_rect(self, x, y, width, height):

cv2.rectangle(self.\_img, (x, y), (x + width, y + height), (0, 255, 0), 2)

def save\_to(self, path):

cv2.imwrite(path, self.\_img)

def to\_numpy\_array(self):

return np.array(self.\_img, dtype=np.uint8).flatten()

def put\_text(self, text, x, y):

cv2.putText(self.\_img, text, (x, y), cv2.FONT\_HERSHEY\_PLAIN, 2,

(0, 255, 0), 2)

def face\_areas(self):

return cv2.CascadeClassifier("/home/aethelbert/Programming/PROGRAMMING/trackery-core-obj-1/eigen-core/haarcascade\_frontalface\_default.xml") \

.detectMultiScale(self.\_img, scaleFactor=1.2, minNeighbors=5,

minSize=(150, 150), flags=cv2.cv.CV\_HAAR\_SCALE\_IMAGE)

def show(self):

cv2.imshow('', self.\_img)

cv2.waitKey(0)

def load\_image\_from(path):

return Image(cv2.imread(path))

class FaceDetector:

def \_\_init\_\_(self):

storage = "/home/aethelbert/Programming/PROGRAMMING/trackery-core-obj-1/public/img/faces"

self.clf = EigenFaces()

self.clf.train("{}/{}".format(storage, "training\_images"))

def show(self, image, wait=True):

for (x, y, w, h) in image.face\_areas():

image.draw\_rect(x, y, w, h)

face = image.cut(x, y, w, h).gray().scale(150, 150).to\_numpy\_array()

oldMean = np.mean(face, 0)

imgState = "Normal"

if oldMean < 125.00:

imgState = "Dark"

if oldMean > 190.00:

imgState = "Bright"

newMean, face = self.fixIllumination(imgState, oldMean, face)

predicted\_name = self.clf.predict\_face(face)

image.put\_text(predicted\_name, x + 20, y + h + 45)

print predicted\_name

def fixIllumination(self, state, oMean, imgF):

newMean = 0

ctr = 0

if state == "Normal":

newMean = oMean

if state == "Bright":

subtr = oMean - 190.00 - 1

for pix in imgF:

if (pix - subtr) >= 0:

imgF[ctr] = pix - subtr

if (pix - subtr) < 0:

imgF[ctr] = 0

ctr = ctr + 1

newMean = np.mean(imgF, 0)

if state == "Dark":

addit = 125.00 - oMean + 1

for pix in imgF:

if (pix + addit) <= 255:

imgF[ctr] = pix + addit

if (pix + addit) > 255:

imgF[ctr] = 255

ctr = ctr + 1

newMean = np.mean(imgF, 0)

return newMean, imgF

**webcam.py**

"""Webcam utilities."""

import cv2

import gfx

KEY\_ESC = 27

def capture():

vc = cv2.VideoCapture(0)

if vc.isOpened():

\_, frame = vc.read()

return gfx.Image(frame)

def display():

vc = cv2.VideoCapture(0)

key = 0

success = True

face\_detector = gfx.FaceDetector()

while success and key != KEY\_ESC:

success, frame = vc.read()

face\_detector.show(gfx.Image(frame), wait=False)

key = cv2.waitKey(20)

**Web Views (Laravel Blade)**

**template.blade.php**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta name="description" content="">

<meta name="author" content="">

<meta name="csrf-token" content="{{ csrf\_token() }}">

<title>@yield('title')</title>

<!-- Bootstrap Core CSS -->

<link rel="stylesheet" type="text/css" href= '/vendor/bootstrap/css/bootstrap.min.css' />

<!-- Theme CSS -->

<link href="css/freelancer.min.css" rel="stylesheet" />

<!-- Custom Fonts -->

<link href="vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css">

<link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet" type="text/css">

<link href="https://fonts.googleapis.com/css?family=Lato:400,700,400italic,700italic" rel="stylesheet" type="text/css">

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->

<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>

<script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>

<![endif]-->

</head>

<body id="page-top" class="index">

<!-- Navigation -->

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top navbar-custom">

<div class="container">

<!-- Brand and toggle get grouped for better mobile display -->

<div class="navbar-header page-scroll">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">

<span class="sr-only">Toggle navigation</span> Menu <i class="fa fa-bars"></i>

</button>

<a class="navbar-brand" href="#page-top">Trackery</a>

</div>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

<ul class="nav navbar-nav navbar-right">

@yield('navbar')

<li class="page-scroll">

<a href="#about">Who Am I?</a>

</li>

<li class="page-scroll">

<a href="#contact">Contact</a>

</li>

</ul>

</div>

<!-- /.navbar-collapse -->

</div>

<!-- /.container-fluid -->

</nav>

<!-- Header -->

<header>

<div class="container">

<div class="row">

<div class="col-lg-12">

@yield('page-image')

<div class="intro-text">

<span class="name">@yield('intro')</span>

@yield('message')

<hr class="star-light">

<span class="skills">FINDING SOMEONE - FOUND SOMEONE - HELP SOMEONE</span>

</div>

</div>

</div>

</div>

</header>

<!-- Content Section -->

@yield('content')

<!-- About Section -->

<section class="success" id="about">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>Who Am I?</h2>

<hr class="star-light">

</div>

</div>

<div class="row">

<div class="col-lg-4 col-lg-offset-2">

<p style="text-align: justify">

Hello mate! I'm Trackery, your virtual assistant for tracing missing people.

</p>

</div>

<div class="col-lg-4">

<p style="text-align: justify">

What are my skills and how can I help you? Just like a normal person I can

recognize someone by remembering their face, in digital term that's what

you called "Face Recognition". Simple enough, I can recognize a person when

someone introduced them to me.

</p>

</div>

</div>

</div>

</section>

<!-- Contact Section -->

<section id="contact">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>Contact Us</h2>

<hr class="star-primary">

</div>

</div>

<div class="row">

<div class="col-lg-8 col-lg-offset-2">

<!-- To configure the contact form email address, go to mail/contact\_me.php and update the email address in the PHP file on line 19. -->

<!-- The form should work on most web servers, but if the form is not working you may need to configure your web server differently. -->

<form name="sentMessage" id="contactForm" novalidate>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>Name</label>

<input type="text" class="form-control" placeholder="Name" id="name" required data-validation-required-message="Please enter your name.">

<p class="help-block text-danger"></p>

</div>

</div>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>Email Address</label>

<input type="email" class="form-control" placeholder="Email Address" id="email" required data-validation-required-message="Please enter your email address.">

<p class="help-block text-danger"></p>

</div>

</div>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>Phone Number</label>

<input type="tel" class="form-control" placeholder="Phone Number" id="phone" required data-validation-required-message="Please enter your phone number.">

<p class="help-block text-danger"></p>

</div>

</div>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>Message</label>

<textarea rows="5" class="form-control" placeholder="Message" id="message" required data-validation-required-message="Please enter a message."></textarea>

<p class="help-block text-danger"></p>

</div>

</div>

<br>

<div id="success"></div>

<div class="row">

<div class="form-group col-xs-12">

<button type="submit" class="btn btn-success btn-lg">Send</button>

</div>

</div>

</form>

</div>

</div>

</div>

</section>

<!-- Footer -->

<footer class="text-center">

<div class="footer-above">

<div class="container">

<div class="row">

<div class="footer-col col-md-6">

<h3>Location</h3>

<p>Gen. Luna Cor. Muralla Street,

<br>Intramuros, Manila, Philippines</p>

</div>

<div class="footer-col col-md-6">

<h3>Developers</h3>

<p>Programmer: <a href="http://facebook.com/edilsalvador">Edilberto D. Salvador Jr.</a>

<br>Researcher: <a href="http://facebook.com/shinzangetsu">Vince Kyle D. Timbol.</a></p>

</div>

</div>

</div>

</div>

<div class="footer-below">

<div class="container">

<div class="row">

<div class="col-lg-12">

Copyright &copy; BertoBros Inc. 2016 - 2017

</div>

</div>

</div>

</div>

</footer>

<!-- Scroll to Top Button (Only visible on small and extra-small screen sizes) -->

<div class="scroll-top page-scroll hidden-sm hidden-xs hidden-lg hidden-md">

<a class="btn btn-primary" href="#page-top">

<i class="fa fa-chevron-up"></i>

</a>

</div>

<!-- jQuery -->

<script src="vendor/jquery/jquery.min.js"></script>

<!-- Bootstrap Core JavaScript -->

<script src="vendor/bootstrap/js/bootstrap.min.js"></script>

<!-- Plugin JavaScript -->

<script src="js/jquery.easing.min.js"></script>

<!-- Contact Form JavaScript -->

<script src="js/jqBootstrapValidation.js"></script>

<script src="js/contact\_me.js"></script>

<!-- Theme JavaScript -->

<script src="js/freelancer.min.js"></script>

</body>

</html>

**home.blade.php**

@extends('template.template')

@section('title')

Trackery - HOME

@endsection

@section('page-image')

<img class="img-responsive" src="img/png/profile.png" alt="trakery">

@endsection

@section('intro')

<small>HI I'M</small><br>TRACKERY

@endsection

@section('navbar')

<li class="hidden">

<a href="#"></a>

</li>

<li class="page-scroll">

<a href="#find">Find</a>

</li>

<li class="page-scroll">

<a href="#found">Found</a>

</li>

@endsection

@section('content')

<!-- Find Section -->

<section id="find">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>FIND</h2>

<hr class="star-primary">

</div>

</div>

<div class="row">

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<p style="text-align: justify">

Are you looking for someone but you're too far from the police station and you didn't know where to call for help?

</p>

</div>

<div class="col-lg-4 col-xs-12">

<p>

I can help you find someone that has been missing.<br>

</p>

</div>

<div class="col-lg-8 col-lg-offset-2 col-xs-12 text-center">

<h3 style="color: #18BC9C">Find them now!</h3>

</div>

<div class="col-lg-8 col-lg-offset-2 col-xs-12 text-center">

<p>My intelligence can help you using my FIND Module.</p>

<p>Click the button below to start.</p>

<a href="/find" class="btn btn-lg btn-primary">

<i class="fa fa-binoculars"></i> Start Finding Someone

</a>

</div>

</div>

</div>

</section>

<!-- Found Section -->

<section class="primary" id="found">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>Found</h2>

<hr class="star-light-primary">

</div>

</div>

<div class="row">

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<p style="text-align: justify">

Found someone missing and you don't know who to approach?

</p>

</div>

<div class="col-lg-4 col-xs-12">

<p style="text-align: justify">

Let me help you connect them to their love ones by registering them to my brain.<br>

</p>

</div>

<div class="col-lg-8 col-lg-offset-2 col-xs-12 text-center">

<h3 style="color: #18BC9C">Report a missing person now!</h3>

</div>

<div class="col-lg-8 col-lg-offset-2 col-xs-12 text-center">

<p>By reporting a missing person you might help someone find their love ones.</p>

<p>Click the button below to start.</p>

<a href="/found" class="btn btn-lg btn-outline">

<i class="fa fa-photo"></i> Report Missing Person

</a>

</div>

</div>

</div>

</section>

@endsection

**found.blade.php**

@extends('template.template')

@section('title')

Trackery - FOUND

@endsection

@section('page-image')

<img class="img-responsive" src="img/png/compass.png" alt="trakery" style="width:20%;height:20%;">

@endsection

@section('intro')

@if(!empty($uploaded))

@if($uploaded == "true")

IMAGE UPLOADED!

@else

NO FACE FOUND!

@endif

@else

FOUND MODULE

@endif

@endsection

@section('message')

@endsection

@section('navbar')

<li class="hidden">

<a href="#"></a>

</li>

<li class="page-scroll">

<a href="#find">Start Uploading</a>

</li>

<li class="page-scroll">

<a href="#directories">Directories</a>

</li>

@endsection

@section('content')

<!-- Find Section -->

<section id="find">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>START UPLOADING</h2>

<hr class="star-primary">

</div>

</div>

<div class="row">

<div class="row">

<div class="col-lg-8 col-lg-offset-2">

@if($result == true)

<div class="col-sm-6 col-xs-12 portfolio-item">

<h3>EXISTING</h3>

<img src={{ $old\_image }} class="img-responsive" alt="Old Image">

</div>

<div class="col-sm-6 col-xs-12 portfolio-item">

<h3>ENHANCED</h3>

<img src={{ $new\_image }} class="img-responsive" alt="New Image">

</div>

&nbsp; <br>

@endif

<form action="found" method="post" enctype="multipart/form-data">

{{-- <input type="hidden" name="\_token" value="{{ csrf\_token() }}"> --}}

{{ csrf\_field() }}

<h3>STEP 1: UPLOAD IMAGE</h3>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>IMAGE</label>

{{-- <input type="hidden" name="MAX\_FILE\_SIZE" value="30000" /> --}}

<input name="image" type="file" required data-validation-required-message="Please upload an image.">

<p class="help-block text-danger"></p>

@if(!empty($valid))

<p class="text-danger">Previous file was invalid. Please enter a valid image file.</p>

@endif

</div>

</div>

<h3>STEP 2: ENTER INFORMATION</h3>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>NAME</label>

<input name="name" type="text" class="form-control" placeholder="Name (if available)" id="name">

<p class="help-block text-danger"></p>

</div>

</div>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>LOCATION</label>

<input name="location" type="text" class="form-control" placeholder="Location" id="location" required data-validation-required-message="Please enter the location where you found them.">

<p class="help-block text-danger"></p>

</div>

</div>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>MESSAGE</label>

<textarea name="message" rows="2" class="form-control" placeholder="Message (optional)" id="message"></textarea>

<p class="help-block text-danger"></p>

</div>

</div>

<br>

<div id="success"></div>

<div class="row">

<div class="form-group col-xs-12">

<button type="submit" class="btn btn-success btn-lg">Send</button>

</div>

</div>

</form>

</div>

</div>

</div>

</div>

</section>

<!-- Found Section -->

<section class="primary" id="directories">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>DIRECTORIES</h2>

<hr class="star-light-primary">

</div>

</div>

<div class="row">

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<center><a href="/home" class="btn btn-lg btn-outline">

<i class="fa fa-home"></i> Home Page

</a></center>

</div>

<div class="col-lg-4 col-xs-12">

<center><a href="/find" class="btn btn-lg btn-outline">

<i class="fa fa-binoculars"></i> &nbsp;Find Page&nbsp;

</a></center>

</div>

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<center><a href="https://www.github.com/PLMCSIT/trackery-core" class="btn btn-lg btn-outline">

<i class="fa fa-github-alt"></i> Github Page

</a></center>

</div>

<div class="col-lg-4 col-xs-12 page-scroll">

<center><a href="#contact" class="btn btn-lg btn-outline">

<i class="fa fa-envelope"></i> Contact Us

</a></center>

</div>

</div>

</div>

</section>

@endsection

**find.blade.php**

@extends('template.template')

@section('title')

Trackery - FINDING

@endsection

@section('page-image')

<img class="img-responsive" src="img/png/map.png" alt="trakery" style="width:20%;height:20%;">

@endsection

@section('intro')

@if(!empty($uploaded))

IMAGE UPLOADED!

@else

FINDING MODULE

@endif

@endsection

@section('navbar')

<li class="hidden">

<a href="#"></a>

</li>

<li class="page-scroll">

<a href="#find">Start Finding</a>

</li>

<li class="page-scroll">

<a href="#directories">Directories</a>

</li>

@endsection

@section('content')

<!-- Find Section -->

<section id="find">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>START FINDING</h2>

<hr class="star-primary">

</div>

</div>

<div class="row">

<div class="row">

<div class="col-lg-8 col-lg-offset-2">

<form action="find" method="post" enctype="multipart/form-data">

{{ csrf\_field() }}

<h3>STEP 1: UPLOAD IMAGE</h3>

<div class="row control-group">

<div class="form-group col-xs-12 floating-label-form-group controls">

<label>IMAGE</label>

{{-- <input type="hidden" name="MAX\_FILE\_SIZE" value="30000" /> --}}

<input name="image" type="file" required data-validation-required-message="Please upload an image.">

<p class="help-block text-danger"></p>

@if(!empty($valid))

<p class="text-danger">Previous file was invalid. Please enter a valid image file.</p>

@endif

</div>

</div>

<br>

<div id="success"></div>

<div class="row">

<div class="form-group col-xs-12">

<button type="submit" class="btn btn-success btn-lg">Send</button>

</div>

</div>

</form>

</div>

</div>

</div>

</div>

</section>

<!-- Directories Section -->

<section class="primary" id="directories">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>DIRECTORIES</h2>

<hr class="star-light-primary">

</div>

</div>

<div class="row">

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<center><a href="/home" class="btn btn-lg btn-outline">

<i class="fa fa-home"></i> Home Page

</a></center>

</div>

<div class="col-lg-4 col-xs-12">

<center><a href="/found" class="btn btn-lg btn-outline">

<i class="fa fa-eye"></i> Found Page

</a></center>

</div>

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<center><a href="https://www.github.com/PLMCSIT/trackery-core" class="btn btn-lg btn-outline">

<i class="fa fa-github-alt"></i> Github Page

</a></center>

</div>

<div class="col-lg-4 col-xs-12 page-scroll">

<center><a href="#contact" class="btn btn-lg btn-outline">

<i class="fa fa-envelope"></i> Contact Us

</a></center>

</div>

</div>

</div>

</section>

@endsection

**train.blade.php**

@extends('template.template')

@section('title')

Trackery - FINDING

@endsection

@section('page-image')

<img class="img-responsive" src=img/png/map.png alt="trakery" style="width:20%;height:20%;">

@endsection

@section('intro')

@if(!empty($uploaded))

IMAGE UPLOADED!

@else

FINDING MODULE

@endif

@endsection

@section('navbar')

<li class="hidden">

<a href="#"></a>

</li>

<li class="page-scroll">

@if(!empty($result))

<a href="#find">Information</a>

@else

<a href="#find">Start Finding</a>

@endif

</li>

<li class="page-scroll">

<a href="#directories">Directories</a>

</li>

@endsection

@section('content')

<!-- Find Section -->

<section id="find">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

@if(!empty($result))

<h2>INFORMATION</h2>

@else

<h2>START FINDING</h2>

@endif

<hr class="star-primary">

</div>

</div>

<div class="row">

<div class="row">

<div class="col-lg-8 col-lg-offset-2">

@if(!empty($result))

<div class="col-sm-6 col-xs-12 portfolio-item">

<h3>Image</h3>

<img src={{ $imagepath }} class="img-responsive" alt="Image">

</div>

<div class="col-sm-6 col-xs-12 portfolio-item">

<h3>Match</h3>

<img src={{ $matchpath }} class="img-responsive" alt="Match">

</div>

&nbsp; <br>

<h3 style="color: #18bc9c">Name:</h3><h2>{{$result[0]->name}}</h2>

<h3 style="color: #18bc9c">Last Seen at:</h3><h2>{{$result[0]->location}}</h2>

<h3 style="color: #18bc9c">Time and Date:</h3><h2>{{$result[0]->created\_at}}</h2>

<h3 style="color: #18bc9c">Other Info:</h3><h2>{{$result[0]->message}}</h2>

<h3 style="color: #18bc9c">System Runtime:</h3><h2>{{$runtime}}</h2>

<br>

<center><a href="/find" class="btn btn-success btn-lg">

<i class="fa fa-photo"></i> Try Again.

</a></center>

@else

<h1>NO IMAGE TO FIND!</h1>

<h3>Upload an image first on found module.</h3>

@endif

</div>

</div>

</div>

</div>

</section>

<!-- Directories Section -->

<section class="primary" id="directories">

<div class="container">

<div class="row">

<div class="col-lg-12 text-center">

<h2>DIRECTORIES</h2>

<hr class="star-light-primary">

</div>

</div>

<div class="row">

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<center><a href="/home" class="btn btn-lg btn-outline">

<i class="fa fa-home"></i> Home Page

</a></center>

</div>

<div class="col-lg-4 col-xs-12">

<center><a href="/found" class="btn btn-lg btn-outline">

<i class="fa fa-eye"></i> Found Page

</a></center>

</div>

<div class="col-lg-4 col-lg-offset-2 col-xs-12">

<center><a href="https://www.github.com/PLMCSIT/trackery-core" class="btn btn-lg btn-outline">

<i class="fa fa-github-alt"></i> Github Page

</a></center>

</div>

<div class="col-lg-4 col-xs-12 page-scroll">

<center><a href="#contact" class="btn btn-lg btn-outline">

<i class="fa fa-envelope"></i> Contact Us

</a></center>

</div>

</div>

</div>

</section>

@endsection

**404.blade.php**

<!DOCTYPE html>

<html>

<head>

<title>Be right back.</title>

<link href="https://fonts.googleapis.com/css?family=Lato:100" rel="stylesheet" type="text/css">

<style>

html, body {

height: 100%;

}

body {

margin: 0;

padding: 0;

width: 100%;

color: #B0BEC5;

display: table;

font-weight: 100;

font-family: 'Lato', sans-serif;

}

.container {

text-align: center;

display: table-cell;

vertical-align: middle;

}

.content {

text-align: center;

display: inline-block;

}

.title {

font-size: 72px;

margin-bottom: 40px;

}

</style>

</head>

<body>

<div class="container">

<div class="content">

<div class="title">

Are you lost?

<a href="{{url('home')}}"><p>Come back to us.</p></a>

</div>

</div>

</div>

</body>

</html>

**503.blade.php**

<!DOCTYPE html>

<html>

<head>

<title>Be right back.</title>

<link href="https://fonts.googleapis.com/css?family=Lato:100" rel="stylesheet" type="text/css">

<style>

html, body {

height: 100%;

}

body {

margin: 0;

padding: 0;

width: 100%;

color: #B0BEC5;

display: table;

font-weight: 100;

font-family: 'Lato', sans-serif;

}

.container {

text-align: center;

display: table-cell;

vertical-align: middle;

}

.content {

text-align: center;

display: inline-block;

}

.title {

font-size: 72px;

margin-bottom: 40px;

}

</style>

</head>

<body>

<div class="container">

<div class="content">

<div class="title">Be right back.</div>

</div>

</div>

</body>

</html>

**Configurations:**

**composer.json**

{

"name": "laravel/laravel",

"description": "The Laravel Framework.",

"keywords": ["framework", "laravel"],

"license": "MIT",

"type": "project",

"require": {

"php": ">=5.6.4",

"laravel/framework": "5.3.\*"

},

"require-dev": {

"fzaninotto/faker": "~1.4",

"mockery/mockery": "0.9.\*",

"phpunit/phpunit": "~5.0",

"symfony/css-selector": "3.1.\*",

"symfony/dom-crawler": "3.1.\*"

},

"autoload": {

"classmap": [

"database"

],

"psr-4": {

"App\\": "app/"

}

},

"autoload-dev": {

"classmap": [

"tests/TestCase.php"

]

},

"scripts": {

"post-root-package-install": [

"php -r \"file\_exists('.env') || copy('.env.example', '.env');\""

],

"post-create-project-cmd": [

"php artisan key:generate"

],

"post-install-cmd": [

"Illuminate\\Foundation\\ComposerScripts::postInstall",

"php artisan optimize"

],

"post-update-cmd": [

"Illuminate\\Foundation\\ComposerScripts::postUpdate",

"php artisan optimize"

]

},

"config": {

"preferred-install": "dist",

"sort-packages": true

}

}

**app.php**

<?php

return [

/\*

|--------------------------------------------------------------------------

| Application Name

|--------------------------------------------------------------------------

|

| This value is the name of your application. This value is used when the

| framework needs to place the application's name in a notification or

| any other location as required by the application or its packages.

\*/

'name' => 'Laravel',

/\*

|--------------------------------------------------------------------------

| Application Environment

|--------------------------------------------------------------------------

|

| This value determines the "environment" your application is currently

| running in. This may determine how you prefer to configure various

| services your application utilizes. Set this in your ".env" file.

|

\*/

'env' => env('APP\_ENV', 'production'),

/\*

|--------------------------------------------------------------------------

| Application Debug Mode

|--------------------------------------------------------------------------

|

| When your application is in debug mode, detailed error messages with

| stack traces will be shown on every error that occurs within your

| application. If disabled, a simple generic error page is shown.

|

\*/

'debug' => env('APP\_DEBUG', true),

/\*

|--------------------------------------------------------------------------

| Application URL

|--------------------------------------------------------------------------

|

| This URL is used by the console to properly generate URLs when using

| the Artisan command line tool. You should set this to the root of

| your application so that it is used when running Artisan tasks.

|

\*/

'url' => env('APP\_URL', 'http://localhost'),

/\*

|--------------------------------------------------------------------------

| Application Timezone

|--------------------------------------------------------------------------

|

| Here you may specify the default timezone for your application, which

| will be used by the PHP date and date-time functions. We have gone

| ahead and set this to a sensible default for you out of the box.

|

\*/

'timezone' => 'UTC',

/\*

|--------------------------------------------------------------------------

| Application Locale Configuration

|--------------------------------------------------------------------------

|

| The application locale determines the default locale that will be used

| by the translation service provider. You are free to set this value

| to any of the locales which will be supported by the application.

|

\*/

'locale' => 'en',

/\*

|--------------------------------------------------------------------------

| Application Fallback Locale

|--------------------------------------------------------------------------

|

| The fallback locale determines the locale to use when the current one

| is not available. You may change the value to correspond to any of

| the language folders that are provided through your application.

|

\*/

'fallback\_locale' => 'en',

/\*

|--------------------------------------------------------------------------

| Encryption Key

|--------------------------------------------------------------------------

|

| This key is used by the Illuminate encrypter service and should be set

| to a random, 32 character string, otherwise these encrypted strings

| will not be safe. Please do this before deploying an application!

|

\*/

'key' => env('APP\_KEY'),

'cipher' => 'AES-256-CBC',

/\*

|--------------------------------------------------------------------------

| Logging Configuration

|--------------------------------------------------------------------------

|

| Here you may configure the log settings for your application. Out of

| the box, Laravel uses the Monolog PHP logging library. This gives

| you a variety of powerful log handlers / formatters to utilize.

|

| Available Settings: "single", "daily", "syslog", "errorlog"

|

\*/

'log' => env('APP\_LOG', 'single'),

'log\_level' => env('APP\_LOG\_LEVEL', 'debug'),

/\*

|--------------------------------------------------------------------------

| Autoloaded Service Providers

|--------------------------------------------------------------------------

|

| The service providers listed here will be automatically loaded on the

| request to your application. Feel free to add your own services to

| this array to grant expanded functionality to your applications.

|

\*/

'providers' => [

/\*

\* Laravel Framework Service Providers...

\*/

Illuminate\Auth\AuthServiceProvider::class,

Illuminate\Broadcasting\BroadcastServiceProvider::class,

Illuminate\Bus\BusServiceProvider::class,

Illuminate\Cache\CacheServiceProvider::class,

Illuminate\Foundation\Providers\ConsoleSupportServiceProvider::class,

Illuminate\Cookie\CookieServiceProvider::class,

Illuminate\Database\DatabaseServiceProvider::class,

Illuminate\Encryption\EncryptionServiceProvider::class,

Illuminate\Filesystem\FilesystemServiceProvider::class,

Illuminate\Foundation\Providers\FoundationServiceProvider::class,

Illuminate\Hashing\HashServiceProvider::class,

Illuminate\Mail\MailServiceProvider::class,

Illuminate\Notifications\NotificationServiceProvider::class,

Illuminate\Pagination\PaginationServiceProvider::class,

Illuminate\Pipeline\PipelineServiceProvider::class,

Illuminate\Queue\QueueServiceProvider::class,

Illuminate\Redis\RedisServiceProvider::class,

Illuminate\Auth\Passwords\PasswordResetServiceProvider::class,

Illuminate\Session\SessionServiceProvider::class,

Illuminate\Translation\TranslationServiceProvider::class,

Illuminate\Validation\ValidationServiceProvider::class,

Illuminate\View\ViewServiceProvider::class,

\Torann\GeoIP\GeoIPServiceProvider::class,

/\*

\* Package Service Providers...

\*/

//

/\*

\* Application Service Providers...

\*/

App\Providers\AppServiceProvider::class,

App\Providers\AuthServiceProvider::class,

// App\Providers\BroadcastServiceProvider::class,

App\Providers\EventServiceProvider::class,

App\Providers\RouteServiceProvider::class,

],

/\*

|--------------------------------------------------------------------------

| Class Aliases

|--------------------------------------------------------------------------

|

| This array of class aliases will be registered when this application

| is started. However, feel free to register as many as you wish as

| the aliases are "lazy" loaded so they don't hinder performance.

|

\*/

'aliases' => [

'App' => Illuminate\Support\Facades\App::class,

'Artisan' => Illuminate\Support\Facades\Artisan::class,

'Auth' => Illuminate\Support\Facades\Auth::class,

'Blade' => Illuminate\Support\Facades\Blade::class,

'Bus' => Illuminate\Support\Facades\Bus::class,

'Cache' => Illuminate\Support\Facades\Cache::class,

'Config' => Illuminate\Support\Facades\Config::class,

'Cookie' => Illuminate\Support\Facades\Cookie::class,

'Crypt' => Illuminate\Support\Facades\Crypt::class,

'DB' => Illuminate\Support\Facades\DB::class,

'Eloquent' => Illuminate\Database\Eloquent\Model::class,

'Event' => Illuminate\Support\Facades\Event::class,

'File' => Illuminate\Support\Facades\File::class,

'Gate' => Illuminate\Support\Facades\Gate::class,

'Hash' => Illuminate\Support\Facades\Hash::class,

'Lang' => Illuminate\Support\Facades\Lang::class,

'Log' => Illuminate\Support\Facades\Log::class,

'Mail' => Illuminate\Support\Facades\Mail::class,

'Notification' => Illuminate\Support\Facades\Notification::class,

'Password' => Illuminate\Support\Facades\Password::class,

'Queue' => Illuminate\Support\Facades\Queue::class,

'Redirect' => Illuminate\Support\Facades\Redirect::class,

'Redis' => Illuminate\Support\Facades\Redis::class,

'Request' => Illuminate\Support\Facades\Request::class,

'Response' => Illuminate\Support\Facades\Response::class,

'Route' => Illuminate\Support\Facades\Route::class,

'Schema' => Illuminate\Support\Facades\Schema::class,

'Session' => Illuminate\Support\Facades\Session::class,

'Storage' => Illuminate\Support\Facades\Storage::class,

'URL' => Illuminate\Support\Facades\URL::class,

'Validator' => Illuminate\Support\Facades\Validator::class,

'View' => Illuminate\Support\Facades\View::class,

'GeoIP' => \Torann\GeoIP\Facades\GeoIP::class,

],

];

**geoip.php**

<?php

return [

/\*

|--------------------------------------------------------------------------

| Logging Configuration

|--------------------------------------------------------------------------

|

| Here you may configure the log settings for when a location is not found

| for the IP provided.

|

\*/

'log\_failures' => true,

/\*

|--------------------------------------------------------------------------

| Include Currency in Results

|--------------------------------------------------------------------------

|

| When enabled the system will do it's best in deciding the user's currency

| by matching their ISO code to a preset list of currencies.

|

\*/

'include\_currency' => true,

/\*

|--------------------------------------------------------------------------

| Default Service

|--------------------------------------------------------------------------

|

| Here you may specify the default storage driver that should be used

| by the framework.

|

| Supported: "maxmind\_database", "maxmind\_api", "ipapi"

|

\*/

'service' => 'maxmind\_database',

/\*

|--------------------------------------------------------------------------

| Storage Specific Configuration

|--------------------------------------------------------------------------

|

| Here you may configure as many storage drivers as you wish.

|

\*/

'services' => [

'maxmind\_database' => [

'class' => \Torann\GeoIP\Services\MaxMindDatabase::class,

'database\_path' => storage\_path('app/geoip.mmdb'),

'update\_url' => 'https://geolite.maxmind.com/download/geoip/database/GeoLite2-City.mmdb.gz',

'locales' => ['en'],

],

'maxmind\_api' => [

'class' => \Torann\GeoIP\Services\MaxMindWebService::class,

'user\_id' => env('MAXMIND\_USER\_ID'),

'license\_key' => env('MAXMIND\_LICENSE\_KEY'),

'locales' => ['en'],

],

'ipapi' => [

'class' => \Torann\GeoIP\Services\IPApi::class,

'secure' => true,

'key' => env('IPAPI\_KEY'),

'continent\_path' => storage\_path('app/continents.json'),

],

],

/\*

|--------------------------------------------------------------------------

| Default Cache Driver

|--------------------------------------------------------------------------

|

| Here you may specify the type of caching that should be used

| by the package.

|

| Options:

|

| all - All location are cached

| some - Cache only the requesting user

| none - Disable cached

|

\*/

'cache' => 'none',

/\*

|--------------------------------------------------------------------------

| Cache Tags

|--------------------------------------------------------------------------

|

| Cache tags are not supported when using the file or database cache

| drivers in Laravel. This is done so that only locations can be cleared.

|

\*/

'cache\_tags' => null,

/\*

|--------------------------------------------------------------------------

| Cache Expiration

|--------------------------------------------------------------------------

|

| Define how long cached location are valid.

|

\*/

'cache\_expires' => 30,

/\*

|--------------------------------------------------------------------------

| Default Location

|--------------------------------------------------------------------------

|

| Return when a location is not found.

|

\*/

'default\_location' => [

'ip' => '127.0.0.0',

'iso\_code' => 'PHP',

'country' => 'Phillipines',

'city' => 'Manila',

'state' => 'NA',

'state\_name' => 'Connecticut',

'postal\_code' => '1007',

'lat' => 14.59,

'lon' => 120.97,

'timezone' => 'America/New\_York',

'continent' => 'NA',

'default' => true,

'currency' => 'PHP',

],

];