Remax Clone

An app for viewing properties, communicating with realtors, reading and commenting on a realtor-managed blog.

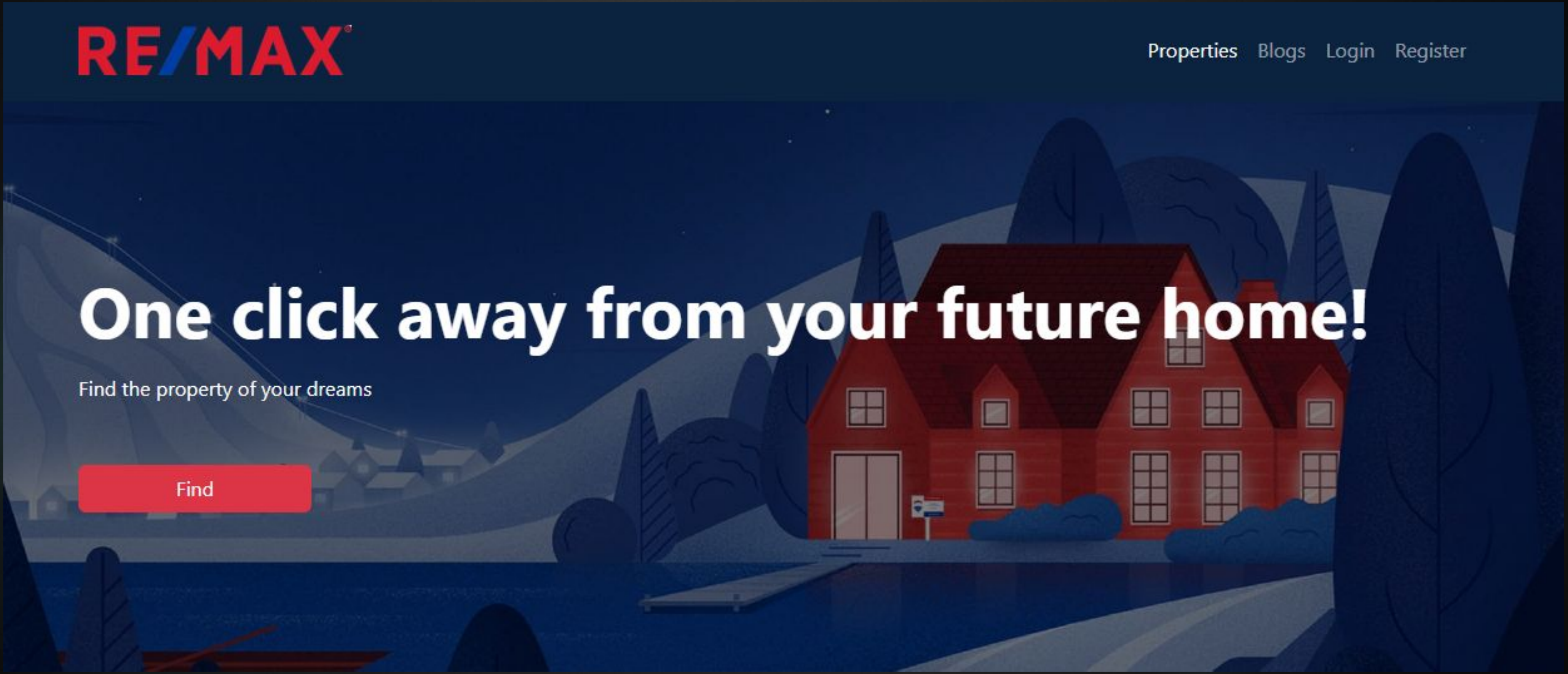
BY
DANG HUYNH MINH (#1670214)
ANASTASSIA TARASSOVA (#1277712)
TRANG TRIEU (#6341464)

Background

- Remax clone is a real estate website.
- Numerous challenging aspects
- Opportunity to reuse the Google Maps API idea from past course
- Will look great in a portfolio

Solution overview

- Home page

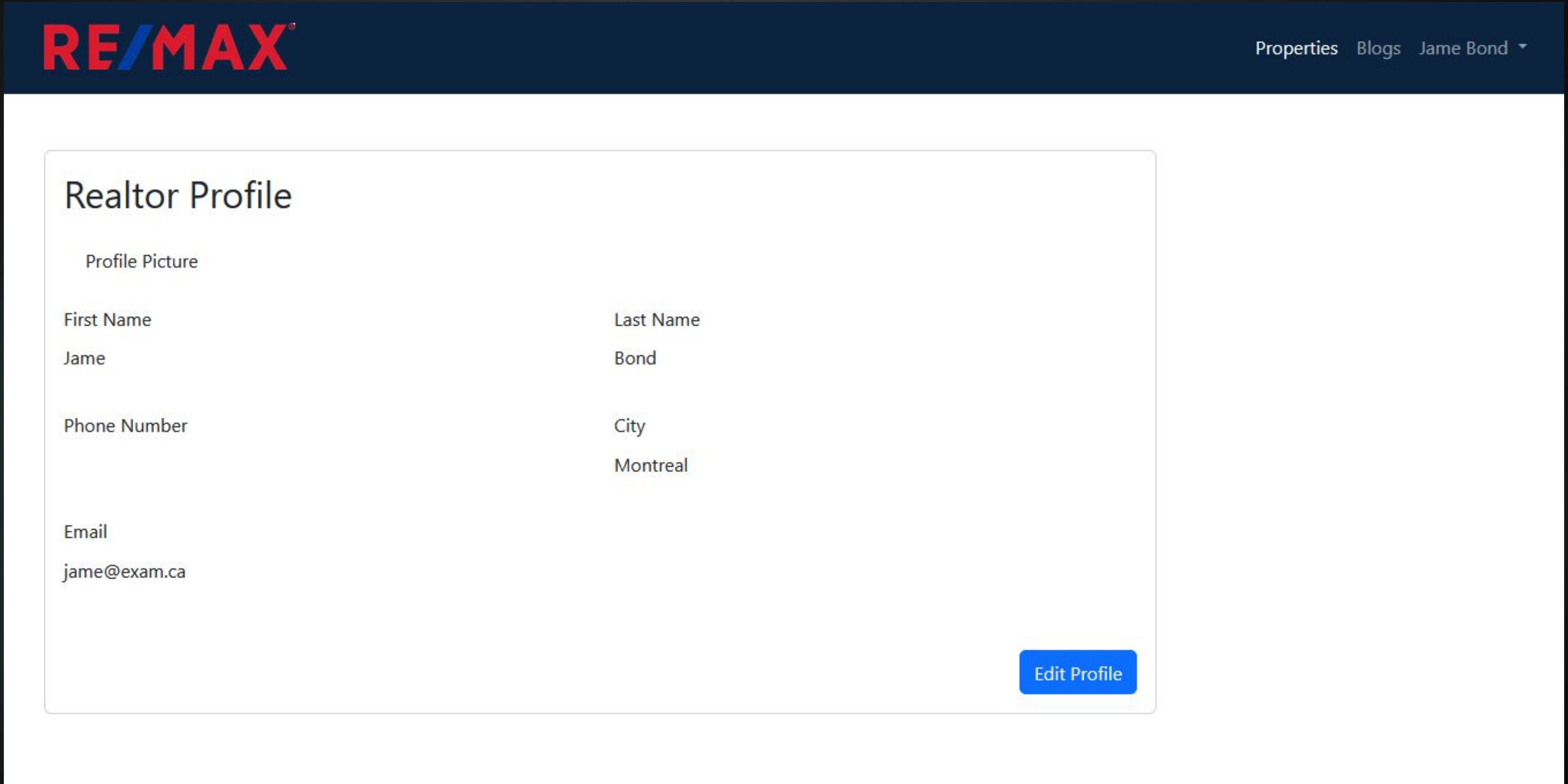


One click away from your future home!

Find the property of your dreams

Solution overview

- User profile



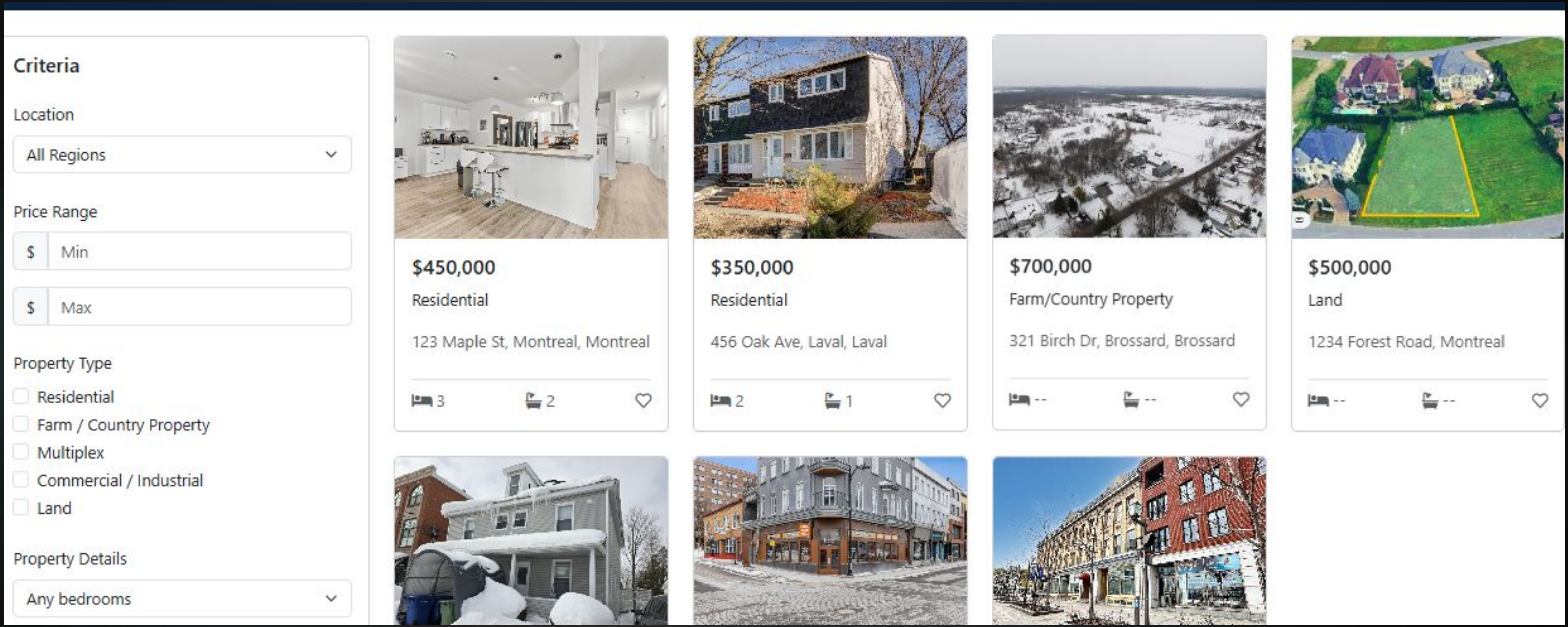
Realtor Profile

Profile Picture

First Name	Last Name
Jame	Bond
Phone Number	City
	Montreal
Email	
jame@exam.ca	

Solution overview

- Search



Criteria

Location

All Regions ⌄

Price Range

$ Min

$ Max

Property Type

- □ Residential
- □ Farm / Country Property
- □ Multiplex
- □ Commercial / Industrial
- □ Land

Property Details

Any bedrooms ⌄

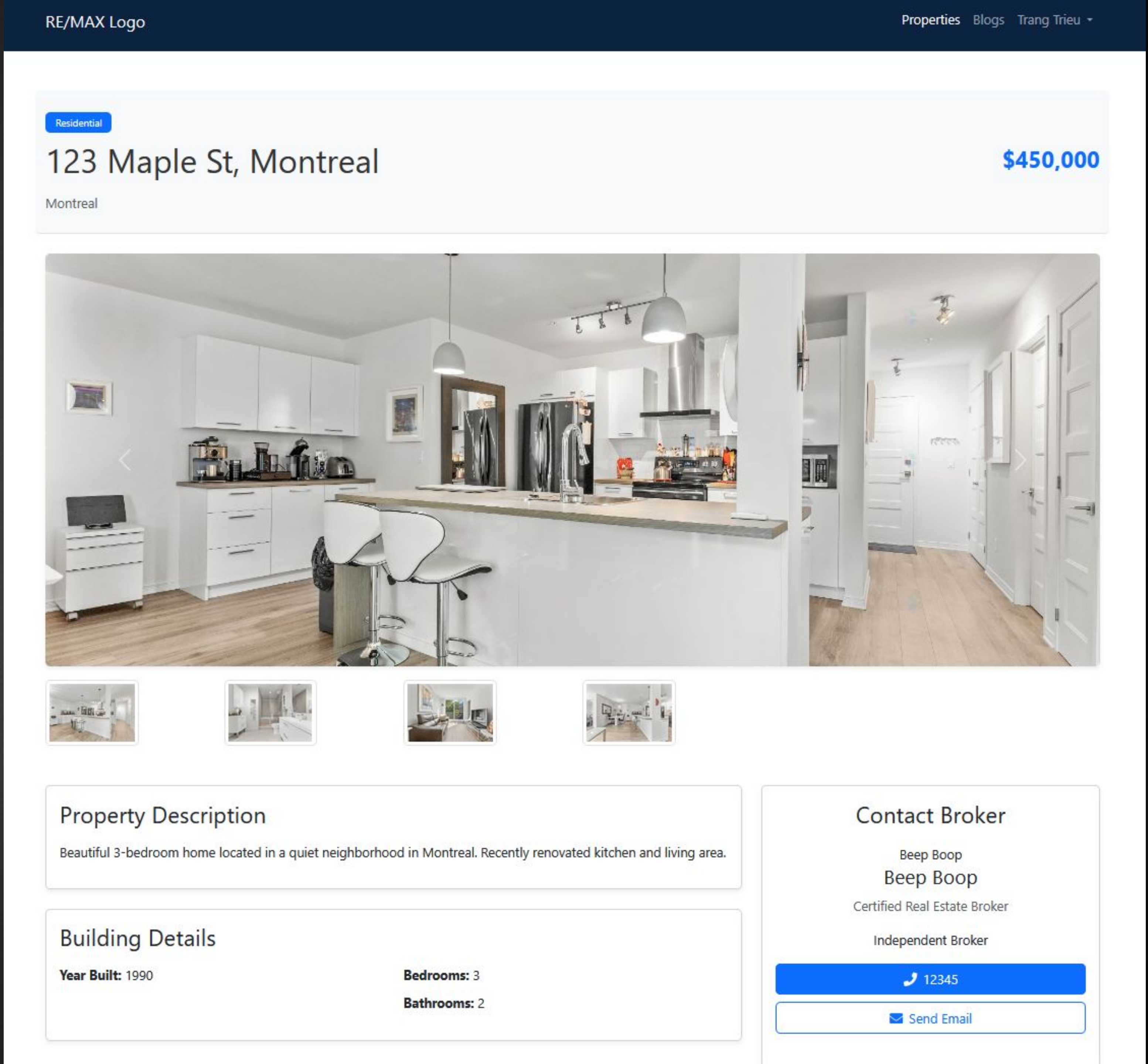
$450,000
Residential
123 Maple St, Montreal, Montreal
🛏 3 🛁 2 ♡

$350,000
Residential
456 Oak Ave, Laval, Laval
🛏 2 🛁 1 ♡

$700,000
Farm/Country Property
321 Birch Dr, Brossard, Brossard
🛏 -- 🛁 -- ♡

$500,000
Land
1234 Forest Road, Montreal
🛏 -- 🛁 -- ♡

Solution overview



Residential

123 Maple St, Montreal

$450,000

Montreal

Property Description

Beautiful 3-bedroom home located in a quiet neighborhood in Montreal. Recently renovated kitchen and living area.

Building Details

Year Built: 1990

Bedrooms: 3

Bathrooms: 2

Contact Broker

Beep Boop

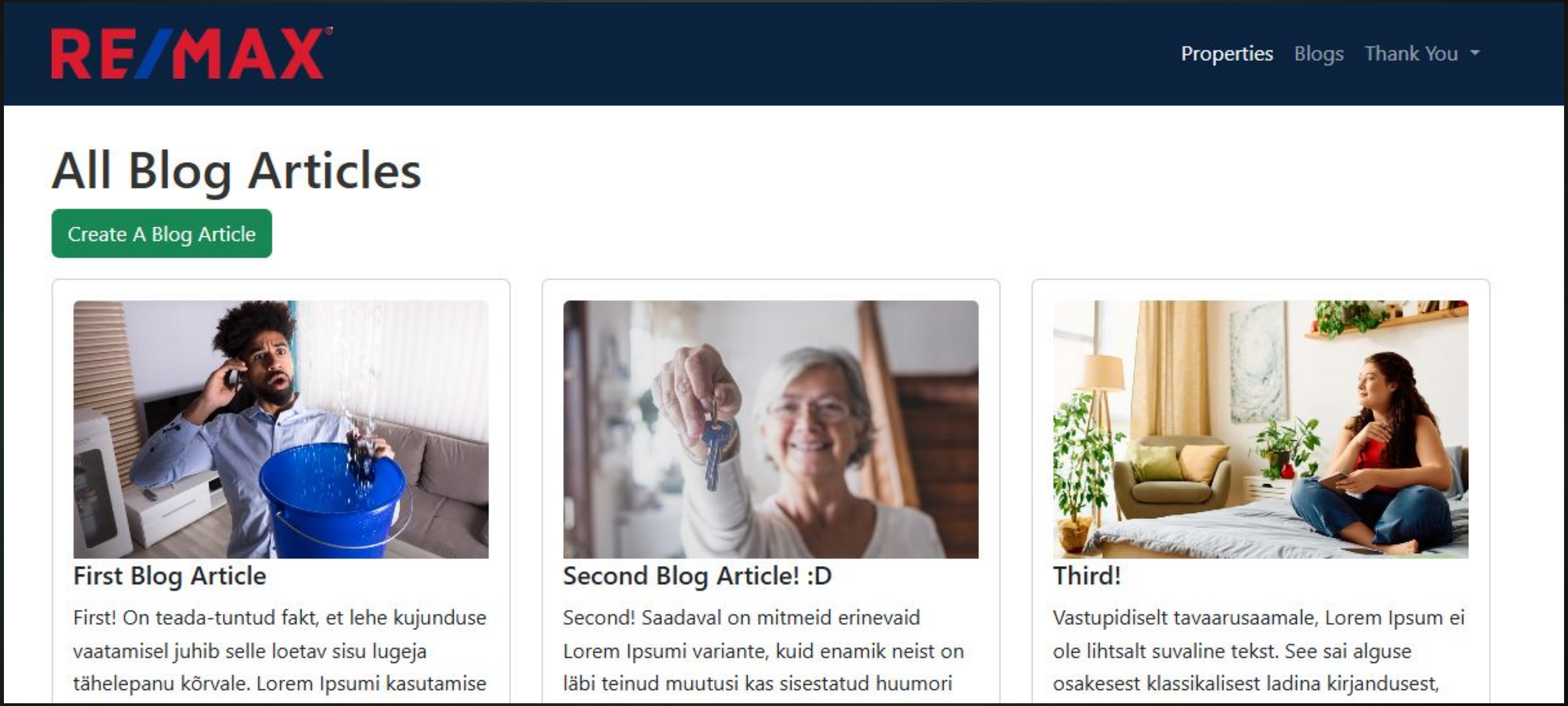
Beep Boop

Certified Real Estate Broker

Independent Broker

Solution overview

- Blogs



All Blog Articles

Create A Blog Article

First Blog Article

First! On teada-tuntud fakt, et lehe kujunduse vaatamisel juhib selle loetav sisu lugeja tähelepanu kõrvale. Lorem Ipsumi kasutamise

Second Blog Article! :D

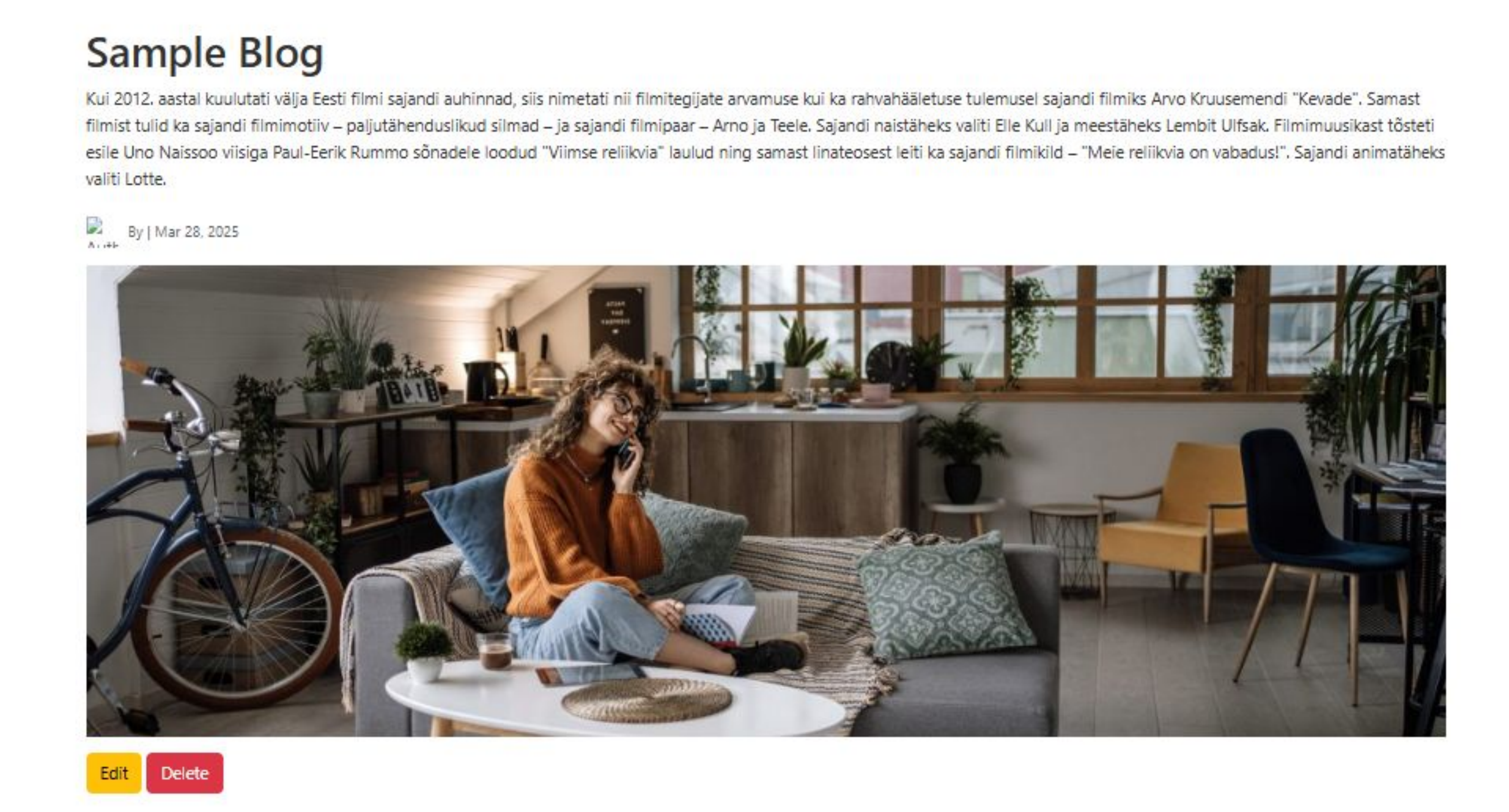
Second! Saadaval on mitmeid erinevaid Lorem Ipsumi variante, kuid enamik neist on läbi teinud muutusi kas sisestatud huumori

Third!

Vastupidiselt tavaarusaamale, Lorem Ipsum ei ole lihtsalt suvaline tekst. See sai alguse osakesest klassikalisest ladina kirjandusest,

Solution overview

- Blog as seen by author (realtor)



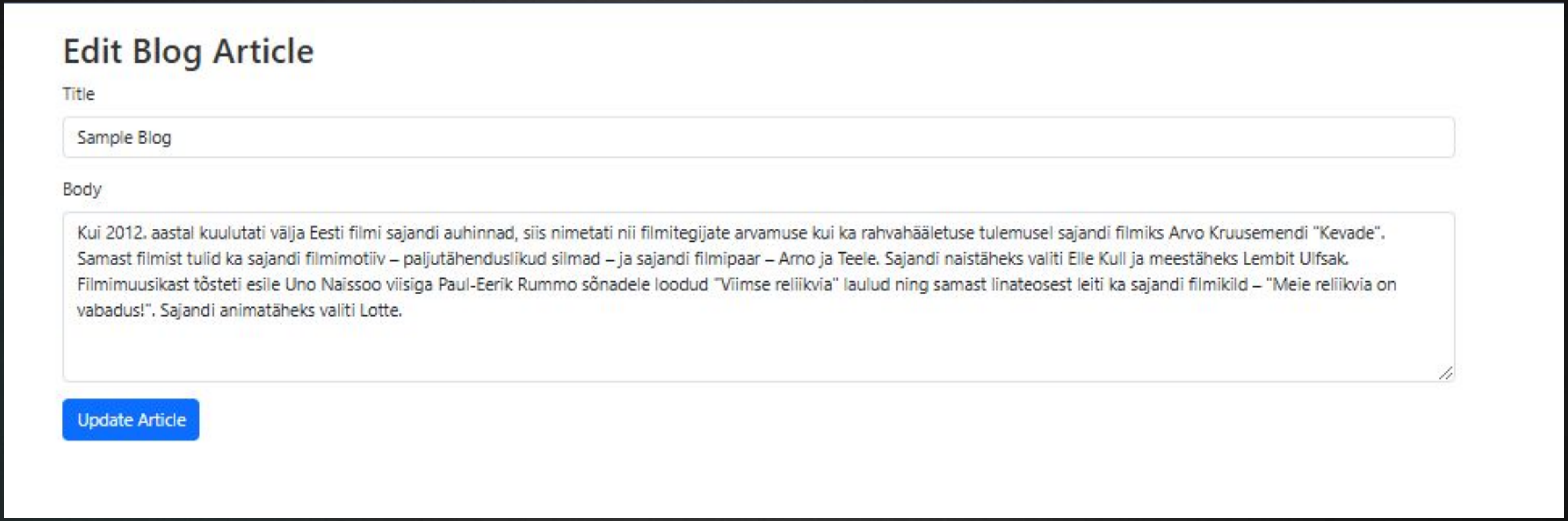
Sample Blog

Kui 2012. aastal kuulutati välja Eesti filmi sajandi auhinnad, siis nimetati nii filmitegijate arvamuse kui ka rahvahääletuse tulemusel sajandi filmiks Arvo Kruusemendi "Kevade". Samast filmist tulid ka sajandi filmimotiiv – paljutähenduslikud silmad – ja sajandi filmipaar – Arno ja Teele. Sajandi naistäheks valiti Elle Kull ja meestäheks Lembit Ulfsak. Filmimuusikast tõsteti esile Uno Naissoo viisiga Paul-Eerik Rummo sõnadele loodud "Viimse reliikvia" laulud ning samast linateosest leiti ka sajandi filmikild – "Meie reliikvia on vabadus!". Sajandi animatäheks valiti Lotte.

By | Mar 28, 2025

Solution overview

- Editing a blog as seen by author



Edit Blog Article

Title

Sample Blog

Body

Kui 2012. aastal kuulutati välja Eesti filmi sajandi auhinnad, siis nimetati nii filmitegijate arvamuse kui ka rahvahääletuse tulemusel sajandi filmiks Arvo Kruusemendi "Kevade". Samast filmist tulid ka sajandi filmimotiiv – paljutähenduslikud silmad – ja sajandi filmipaar – Arno ja Teele. Sajandi naistäheks valiti Elle Kull ja meestäheks Lembit Ulfsak. Filmimuusikast tõsteti esile Uno Naissoo viisiga Paul-Eerik Rummo sõnadele loodud "Viimse reliikvia" laulud ning samast linateosest leiti ka sajandi filmikild – "Meie reliikvia on vabadus!". Sajandi animatäheks valiti Lotte.

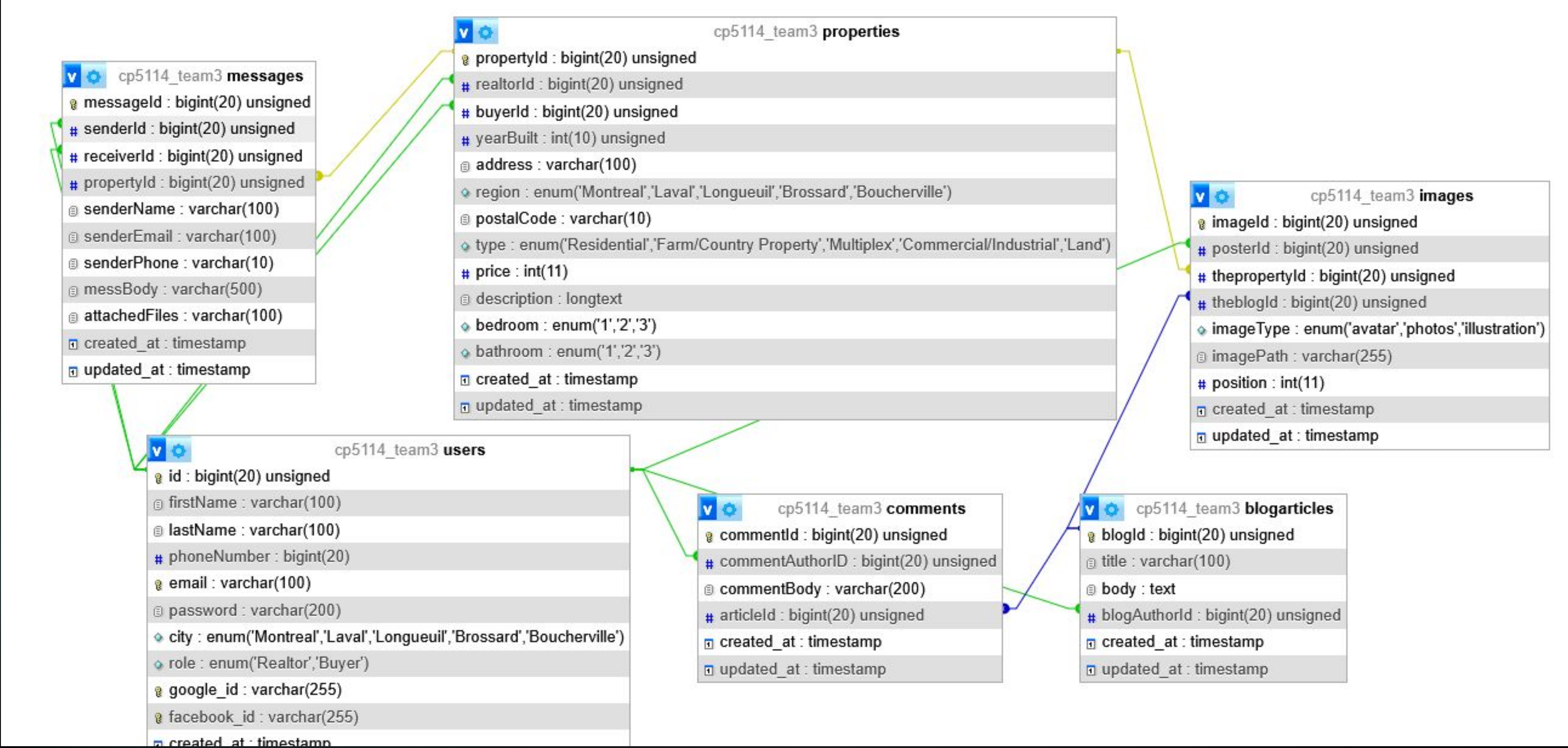
Roles

Dang: profiles

Ana: routes, models, blog, comments

Trang: user login/registration , DB schema, properties

Everyone: front-end, testing



cp5114_team3 **messages**

messageId : bigint(20) unsigned
senderId : bigint(20) unsigned
receiverId : bigint(20) unsigned
propertyId : bigint(20) unsigned
senderName : varchar(100)
senderEmail : varchar(100)
senderPhone : varchar(10)
messBody : varchar(500)
attachedFiles : varchar(100)
created_at : timestamp
updated_at : timestamp

cp5114_team3 **properties**

propertyId : bigint(20) unsigned
realtorId : bigint(20) unsigned
buyerId : bigint(20) unsigned
yearBuilt : int(10) unsigned
address : varchar(100)
region : enum('Montreal','Laval','Longueuil','Brossard','Boucherville')
postalCode : varchar(10)
type : enum('Residential','Farm/Country Property','Multiplex','Commercial/Industrial','Land')
price : int(11)
description : longtext
bedroom : enum('1','2','3')
bathroom : enum('1','2','3')
created_at : timestamp
updated_at : timestamp

cp5114_team3 **images**

imageId : bigint(20) unsigned
posterId : bigint(20) unsigned
thepropertyId : bigint(20) unsigned
theblogId : bigint(20) unsigned
imageType : enum('avatar','photos','illustration')
imagePath : varchar(255)
position : int(11)
created_at : timestamp
updated_at : timestamp

cp5114_team3 **users**

id : bigint(20) unsigned
firstName : varchar(100)
lastName : varchar(100)
phoneNumber : bigint(20)
email : varchar(100)
password : varchar(200)
city : enum('Montreal','Laval','Longueuil','Brossard','Boucherville')
role : enum('Realtor','Buyer')
google_id : varchar(255)
facebook_id : varchar(255)
created_at : timestamp

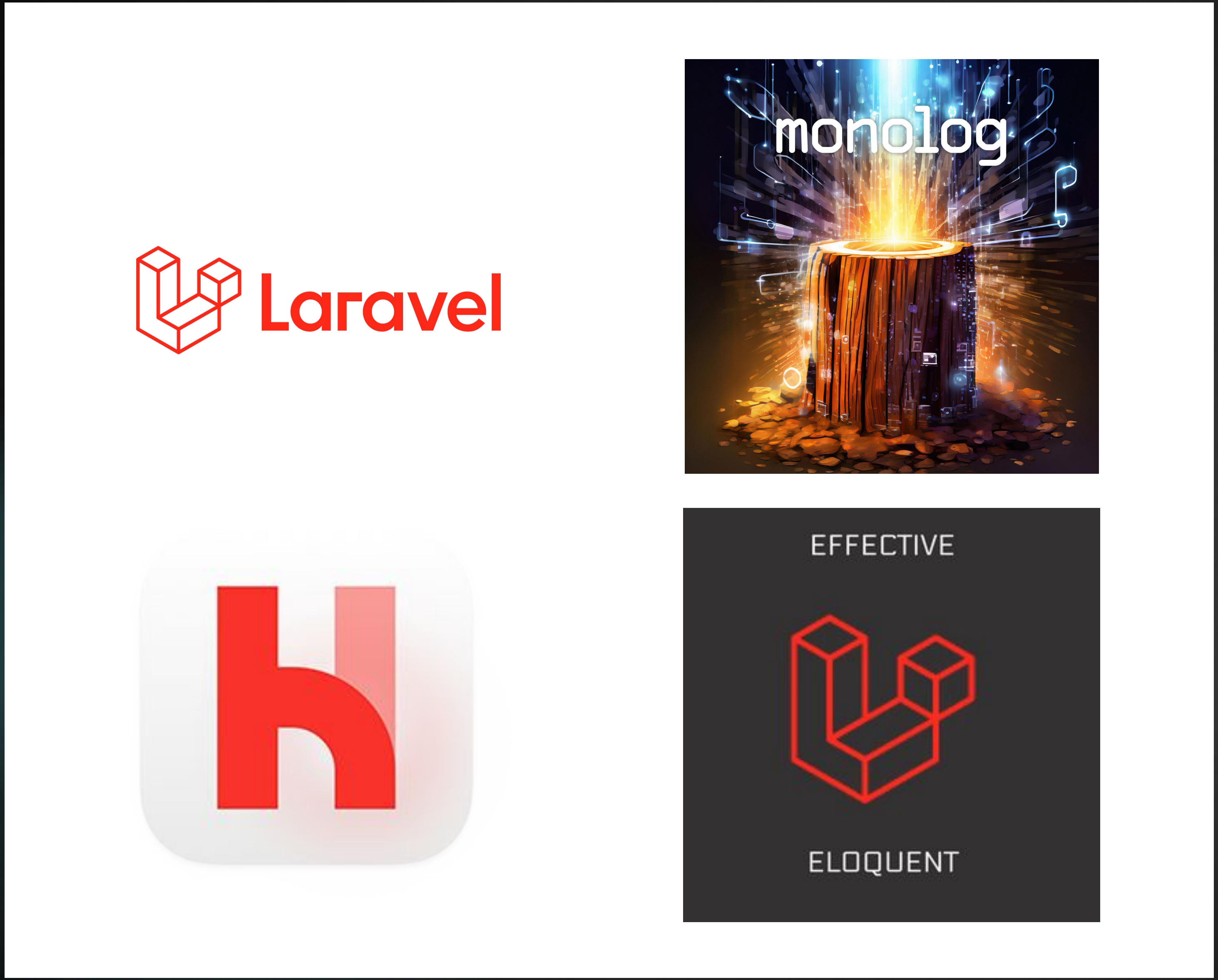
cp5114_team3 **comments**

commentId : bigint(20) unsigned
commentAuthorID : bigint(20) unsigned
commentBody : varchar(200)
articleId : bigint(20) unsigned
created_at : timestamp
updated_at : timestamp

cp5114_team3 **blogarticles**

blogId : bigint(20) unsigned
title : varchar(100)
body : text
blogAuthorId : bigint(20) unsigned
created_at : timestamp
updated_at : timestamp

Database structure

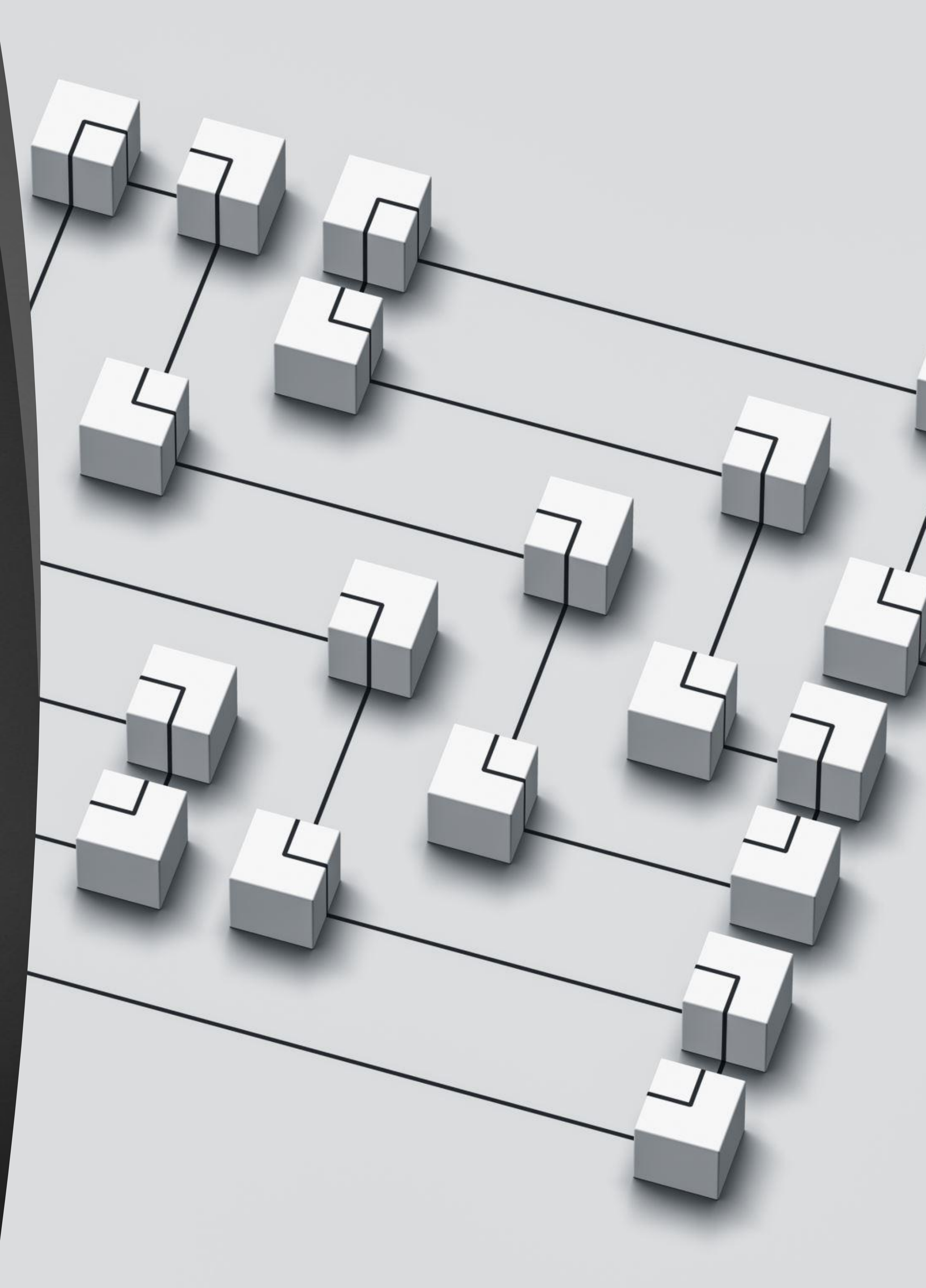


Challenges and Solutions

Challenges

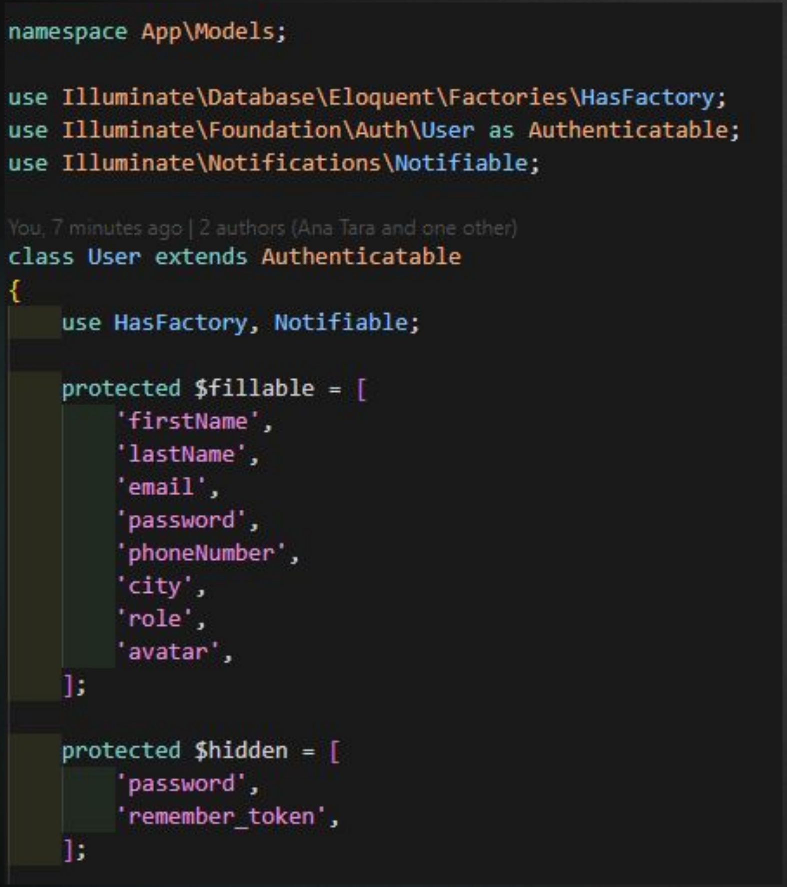
- Technology:
  - Familiarize Laravel framework
  - Switch to Herd from Xamp Mamp Wamp
  - MeekroDB conflicted with Eloquent ORM

- Project
  - Property Search and Filtering
  - Security and Data Protection



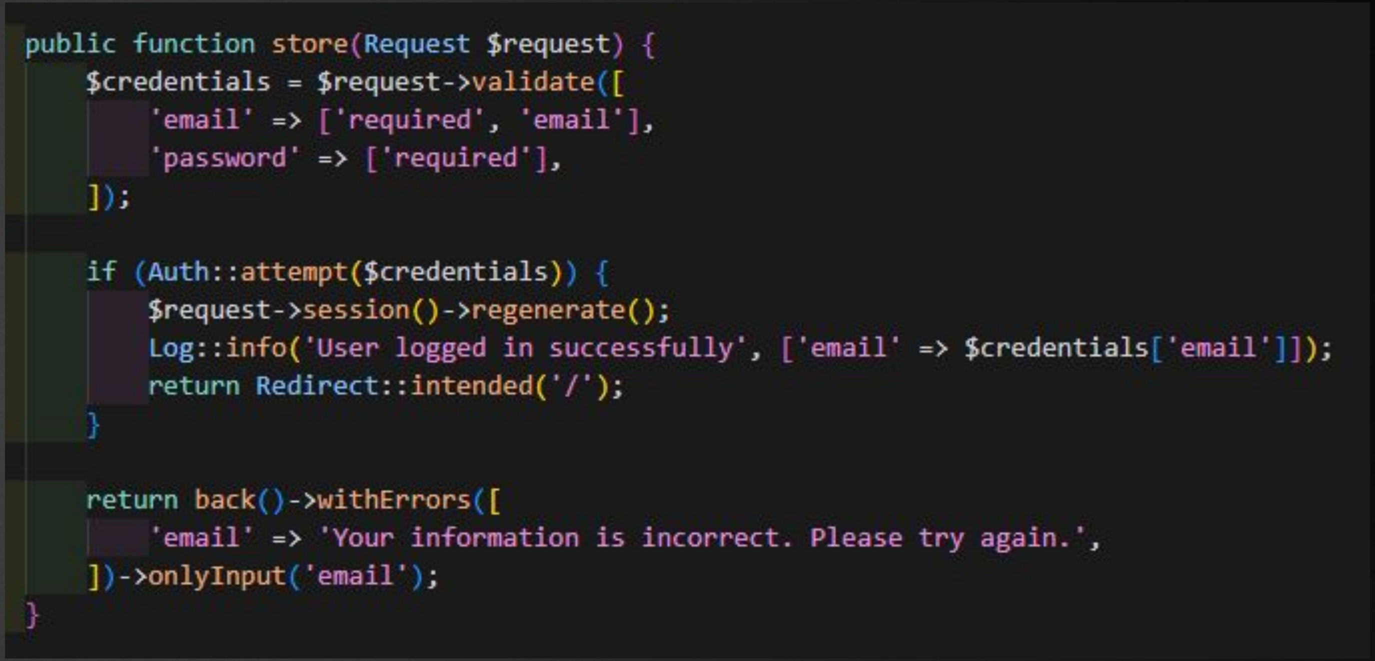
What we learned (Trang)

Login Auth in Laravel



```php
namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;

You, 7 minutes ago | 2 authors (Ana Tara and one other)
class User extends Authenticatable
{
    use HasFactory, Notifiable;

    protected $fillable = [
        'firstName',
        'lastName',
        'email',
        'password',
        'phoneNumber',
        'city',
        'role',
        'avatar',
    ];

    protected $hidden = [
        'password',
        'remember_token',
    ];
```

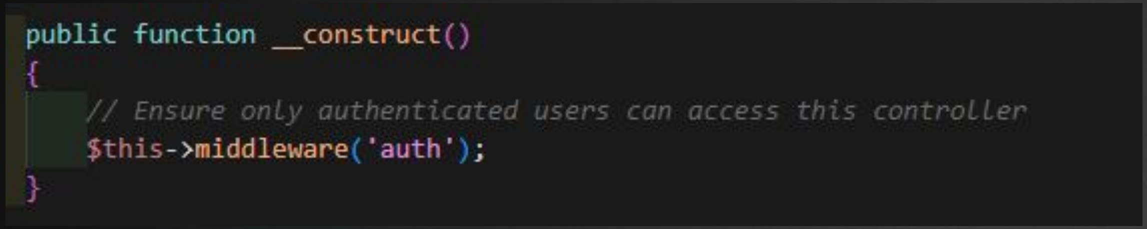
User model



```php
public function store(Request $request) {
    $credentials = $request->validate([
        'email' => ['required', 'email'],
        'password' => ['required'],
    ]);

    if (Auth::attempt($credentials)) {
        $request->session()->regenerate();
        Log::info('User logged in successfully', ['email' => $credentials['email']]);
        return Redirect::intended('/');
    }

    return back()->withErrors([
        'email' => 'Your information is incorrect. Please try again.',
    ])->onlyInput('email');
}
```

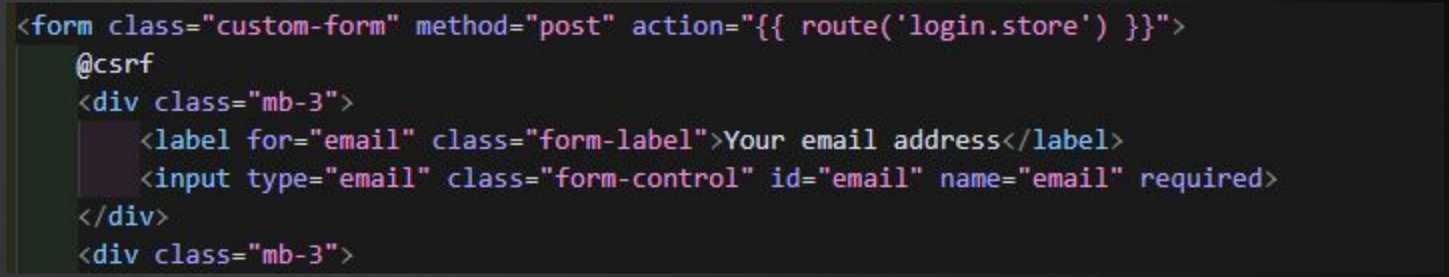
Login Controller

What we learned



```php
public function __construct()
{
    // Ensure only authenticated users can access this controller
    $this->middleware('auth');
}
```

Profile Controller



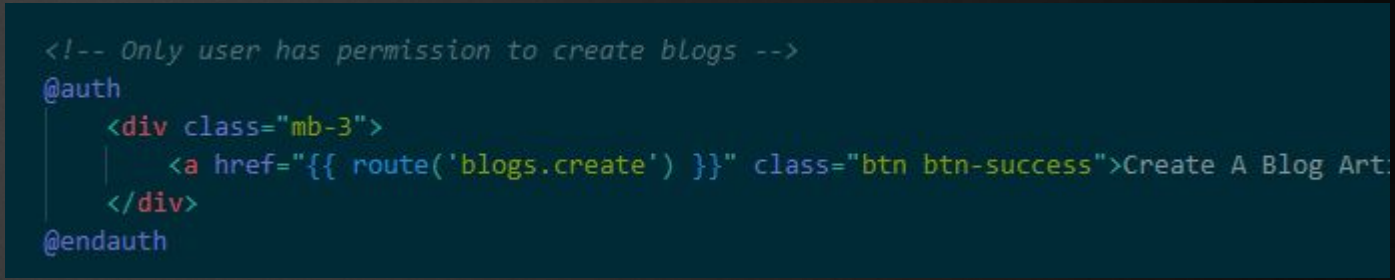
```html
<form class="custom-form" method="post" action="{{ route('login.store') }}">
    @csrf
    <div class="mb-3">
        <label for="email" class="form-label">Your email address</label>
        <input type="email" class="form-control" id="email" name="email" required>
    </div>
    <div class="mb-3">
```

Login View
(login.blade.php)

What we learned (Ana)

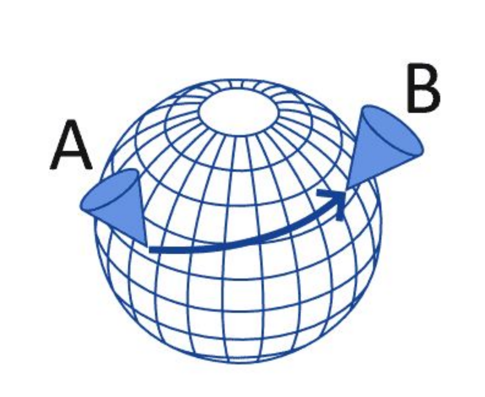
- Laravel
  Eloquent ORM

- @guest and @auth

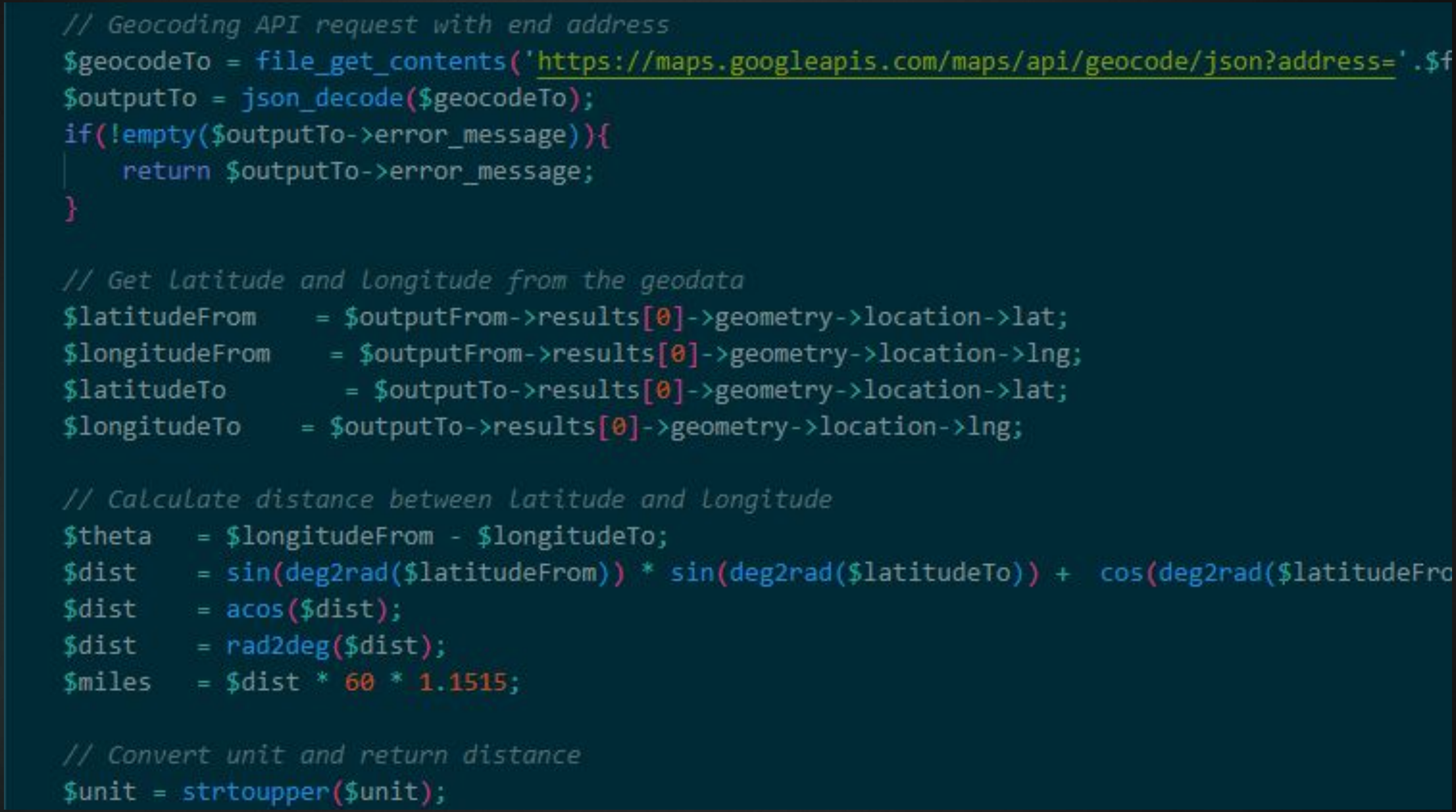


```
<!-- Only user has permission to create blogs -->
@auth
    <div class="mb-3">
        <a href="{{ route('blogs.create') }}" class="btn btn-success">Create A Blog Art
    </div>
@endauth
```

- Haversine formula



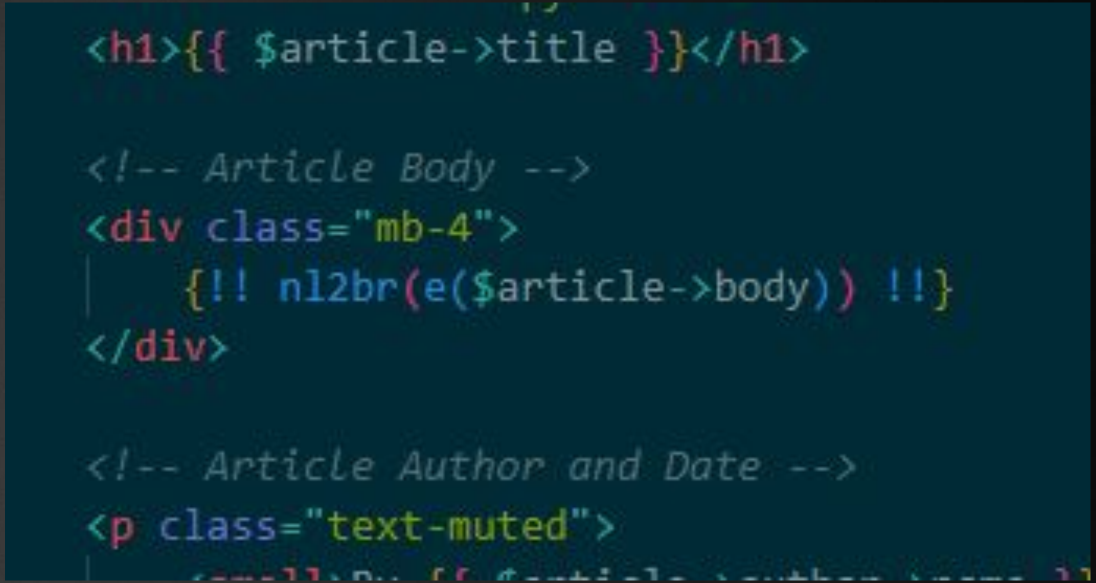
Haversine formula



```
// Geocoding API request with end address
$geocodeTo = file_get_contents('https://maps.googleapis.com/maps/api/geocode/json?address='.$f
$outputTo = json_decode($geocodeTo);
if(!empty($outputTo->error_message)){
    return $outputTo->error_message;
}

// Get latitude and longitude from the geodata
$latitudeFrom    = $outputFrom->results[0]->geometry->location->lat;
$longitudeFrom    = $outputFrom->results[0]->geometry->location->lng;
$latitudeTo        = $outputTo->results[0]->geometry->location->lat;
$longitudeTo    = $outputTo->results[0]->geometry->location->lng;

// Calculate distance between latitude and longitude
$theta    = $longitudeFrom - $longitudeTo;
$dist    = sin(deg2rad($latitudeFrom)) * sin(deg2rad($latitudeTo)) +  cos(deg2rad($latitudeFro
$dist    = acos($dist);
$dist    = rad2deg($dist);
$miles    = $dist * 60 * 1.1515;

// Convert unit and return distance
$unit = strtoupper($unit);
```

What we learned (Ana)

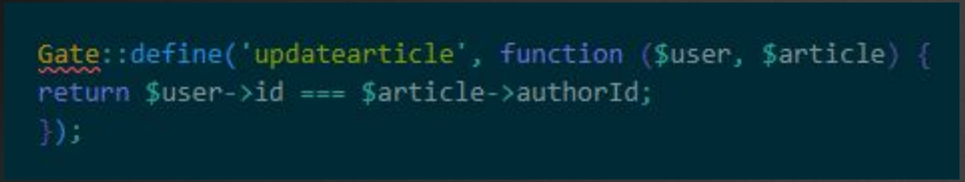
- Escape any potentially harmful HTML/JS using e()
  - <script>alert('XSS')</script> becomes &lt;script&gt;alert('XSS')&lt;/script&gt;
- Convert newline characters in the article body to actual HTML line breaks using nl2br().
  - two lines become line1<br>line2
- Render this safe HTML back into the page using {!! !!}
  - Used only for safe HTML



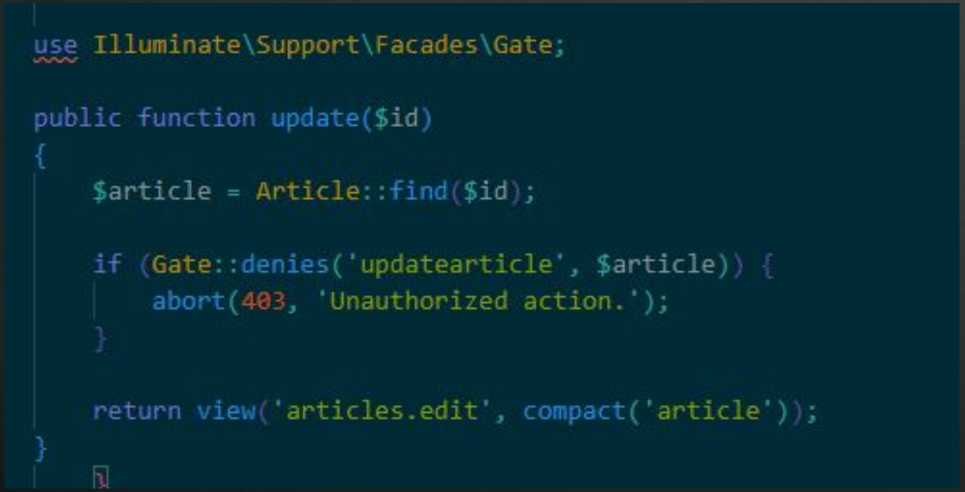
```
<h1>{{ $article->title }}</h1>

<!-- Article Body -->
<div class="mb-4">
    {!! nl2br(e($article->body)) !!}
</div>

<!-- Article Author and Date -->
<p class="text-muted">
```

What we learned (Ana)

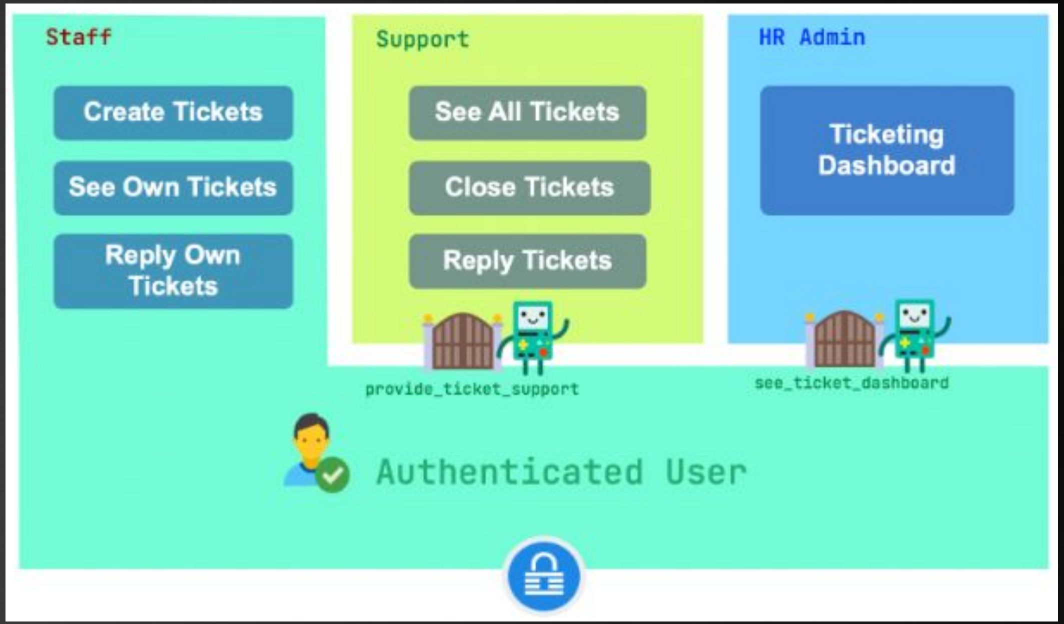
- Laravel Gates
- Defined in AuthServiceProvider
- Can be associated with façades



```php
Gate::define('updatearticle', function ($user, $article) {
return $user->id === $article->authorId;
});
```

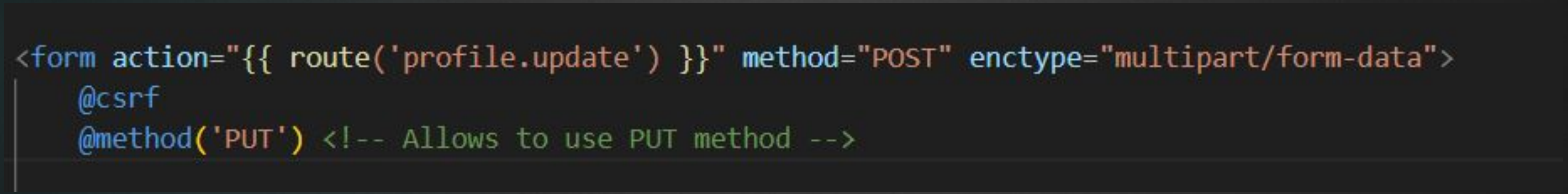


```php
use Illuminate\Support\Facades\Gate;

public function update($id)
{
    $article = Article::find($id);

    if (Gate::denies('updatearticle', $article)) {
        abort(403, 'Unauthorized action.');
    }

    return view('articles.edit', compact('article'));
}
```



What we learned (Dang)

- @csrf
- Put method in form



```
<form action="{{ route('profile.update') }}" method="POST" enctype="multipart/form-data">
    @csrf
    @method('PUT') <!-- Allows to use PUT method -->
```

Future work

- Finalize internal messaging system
- Finalize Google Maps Geolocalisation API
- Walking distance from amenities (schools, hospitals, shopping centers, parks, etc.)
- Comments should be moderated via Edit and Delete
- Autosuggestions based on past searches
- Arrays of liked properties by users

Summary

- *We built a Remax clone website for realtors and potential buyers.*

Users can register, login and view their profile.

Users can search and view listed properties.

Users can read blog articles and comments.

Registered users can comment on blog posts.

Realtors can CRUD properties and blog posts.

Some features are missing/not completed.

Each feature functions while being loosely connected to each other.

Thank you for listening

988 SUICIDE & CRISIS LIFELINE

24/7 CALL, TEXT, CHAT