

Invest Idea App

Team: Idea Market
올다세브 기요실딘
하크베르디에브 발리존

Contents



01. Introduce

- 팀원 소개
- Invest Idea란?



02. Process

- Frontend 구성
- Backend 구성



03. Conclusion

- 결론
- 웹 페이지



프로젝트팀 소개

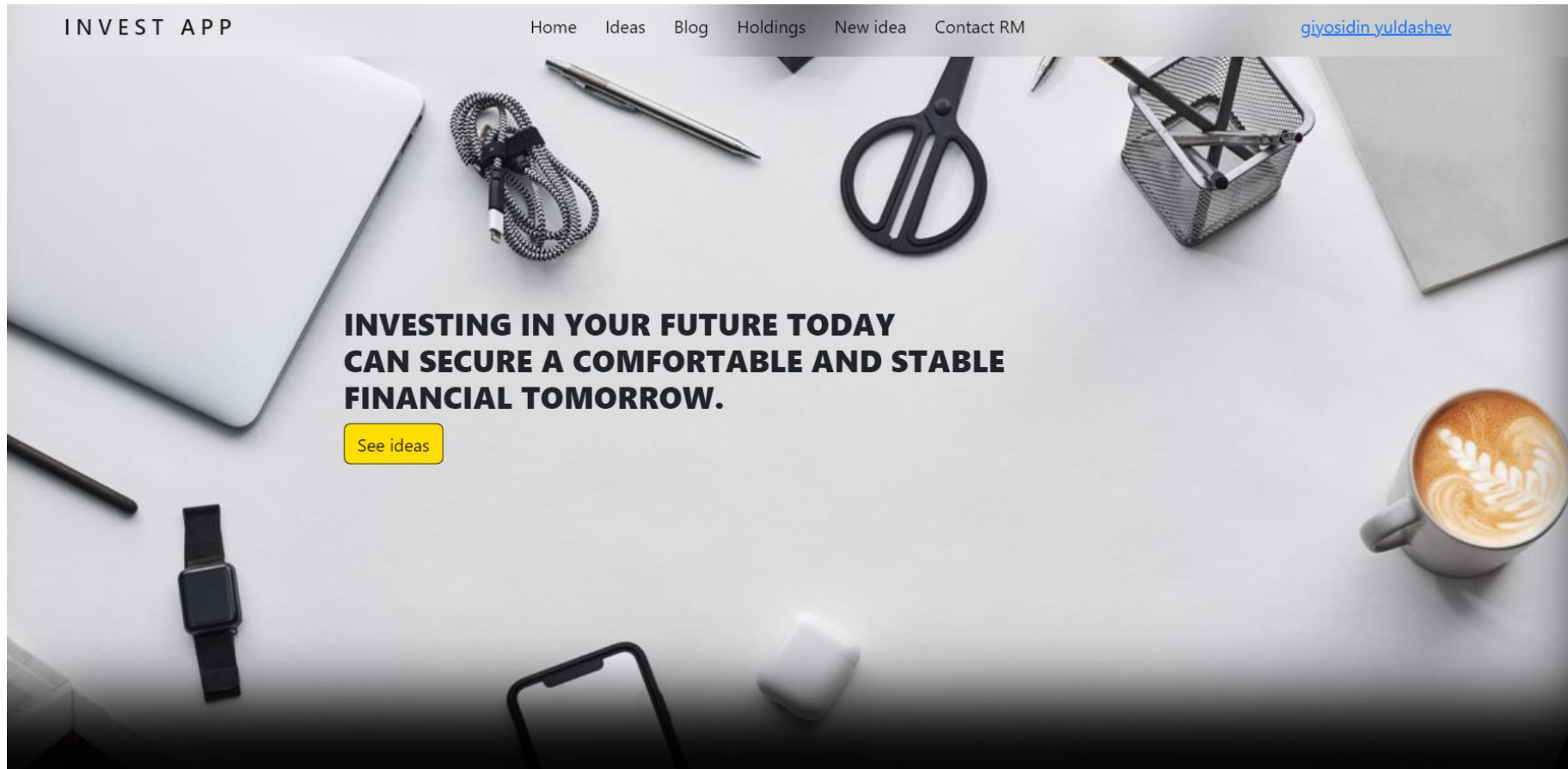


팀장: 율다세브 기요실딘
(Frontend, Backend, ppt)



팀원: 하크베르디에브 발리존
(Design, Idea)

Invest Idea란?



이 프로젝트의 목표는 좋은 사업 아이디어를 가진 사람들과 사업에 돈을 투자할 수 있는 사람들을 만날 수 있게 하는 것입니다.



Invest Idea란?

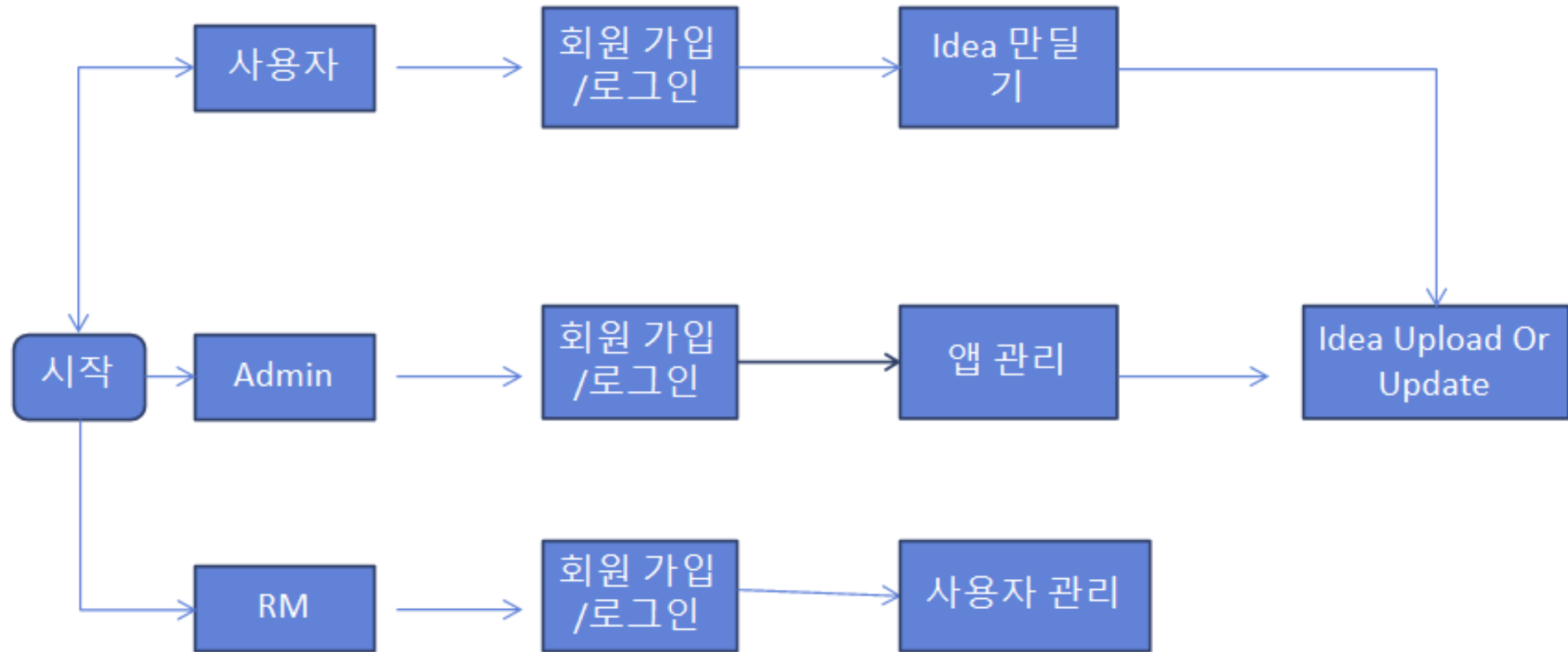
Idea있고 돈 없는 분.



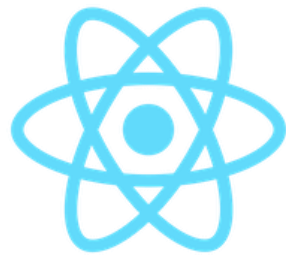
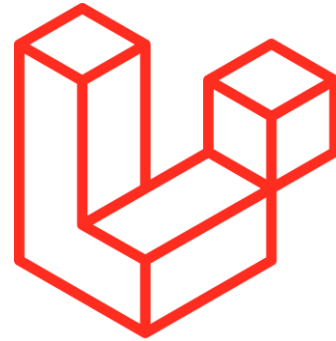
돈 있고 Idea가 없는 분.



시스템 구성

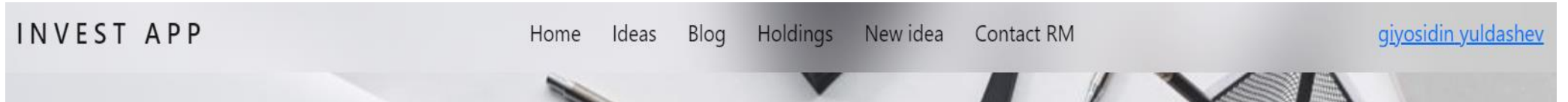


Frontend & Backend



Frontend 구성

홈페이지, 내비게이션바



Home: 홈페이지로 이동한다

Ideas: Idea 모임 페이지로 이동한다.

Blog: 블로그 및 뉴스페이지로 이동한다.

Holdings: 계약 페이지로 이동한다.

New Idea: 새로운 Idea를 등록하는 페이지로 이동한다.

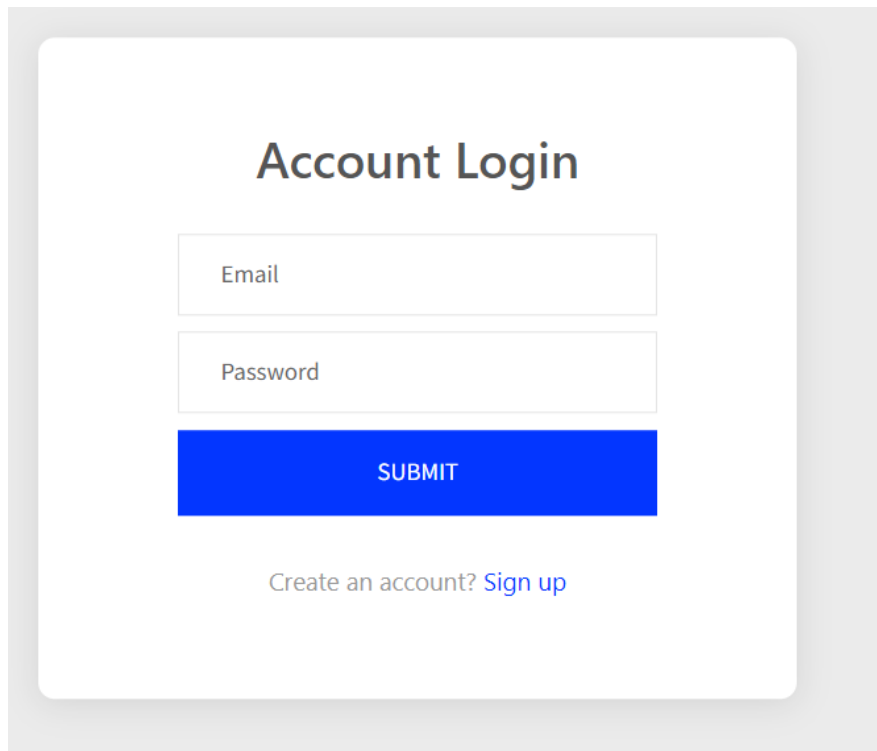
Contact RM: RM 연락 페이지로 이동한다.

사용자이름: 사용자 페이지로 이동한다.

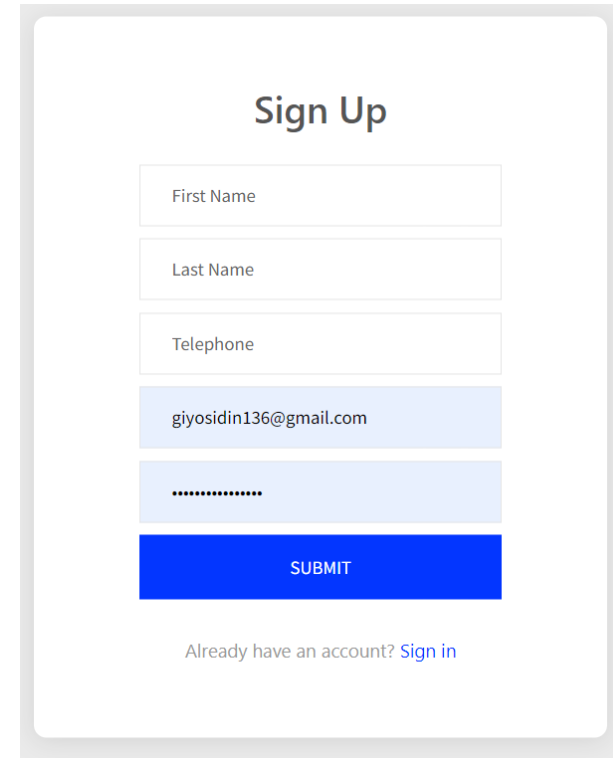


Frontend 구성

Account.
Sign In & Sign Up.



The image shows a web form titled "Account Login". It features two input fields: "Email" and "Password". Below these fields is a prominent blue "SUBMIT" button. At the bottom of the form, there is a link that says "Create an account? Sign up".



The image shows a web form titled "Sign Up". It contains four input fields: "First Name", "Last Name", "Telephone", and "Email". The "Email" field is pre-filled with "giyosidin136@gmail.com". Below the "Telephone" field is a field for a password, represented by a series of dots. A blue "SUBMIT" button is located below the password field. At the bottom of the form, there is a link that says "Already have an account? Sign in".

FirstName, LastName, Telephone, Email, Password를 작성하고 회원 가입이 가능합니다.
이후에 Sign In 할 수 있습니다.



[All ideas](#) / Recommended ideas by RM

Shopping Mall

Shopping Mall for Russia

Risk rating: 10/9

Instruments: Shopping Mall,

Currency: 2000000 RUB,

Major Sector: Seller major

Minor Sector: Counterman

Region: Europe,

Country:  Russia,

Publish date: 12/17/2023, 7:43 PM **Expiry Date:** 12/14/2023, 7:43 PM

Author: giyosidin yuldashev

[READ MORE](#)

이 페이지는 모든 Idea 페이지입니다. 여기서 원하는 것 골라서 계약할 수 있습니다.



Frontend 구성

Holdings

ST APP

[Home](#) [Ideas](#) [Blog](#) [Holdings](#) [New idea](#) [Contact RM](#)

[giyosidin.yuldashev](#)

Holdings list

You have an empty holdings list. Please become an investor! 💰

Contact me

giyosidin136@gmail.com

어떤 아이디어에 투자하면 Holdings List 목록에 나타납니다.



Frontend 구성

New Idea.

INVEST APP

[Home](#) [Ideas](#) [Blog](#) [Holdings](#) [New idea](#) [Contact RM](#)

[giyosidin.yuldashev](#)

Send your idea

Name

Email

Message

SUBMIT

Contact me

giyosidin136@gmail.com

새로운 아이디어가 있으면 New Idea 섹션을 클릭하고 아이디어를 등록할 수 있습니다.
새로운 아이디어는 관리자만 볼 수 있습니다.



Frontend 구성

Contact RM.

T A P P

Home Ideas Blog Holdings New idea Contact RM

RMs list

user5 user5
user5@mail.com 01024057772
Call

RM은 사이트에서 사용자의 프로필을 볼 수 있으며, 좋은 사업가에게 최고의 아이디어를 판매합니다.



Frontend 구성

프로필 수정 페이지

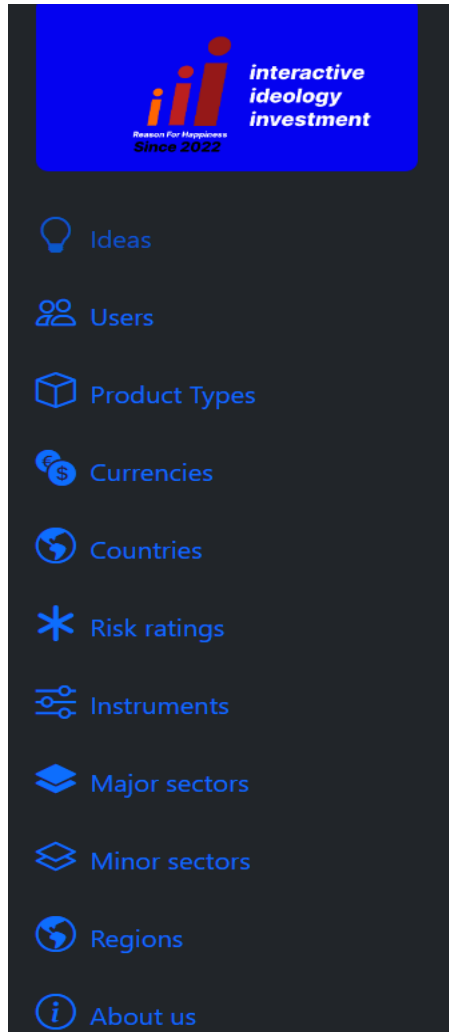
Profile settings

Credentials

FirstName, LastName, Telephone, Email, Password 이를 작성 하고 CHANGE 버튼 눌러면 수정이 가능합니다.



Frontend 구성, Admin Panel



Ideas list

Add new idea

Admin만 사이트에 모든 새로운 아이디어를 추가할 수 있습니다.

Users 에 모든 사용자들 보고 관리할 수 있습니다.

Product types list에서 제품 유형에 대한 정보가 목록에 기록됩니다.

Currencies list에서 아이디어에 들어갈 자금을 입력합니다.

Countries list에서 그 아이디어를 어느 나라에서 사용할 것인가를 입력합니다.

기타 Risk ratings, Instruments, Major sectors, Minor sectors, Regions 와 같은 정보들을 입력하면 새 Idea가 Upload 됩니다.

Frontend 구성, PHP MyAdmin

The screenshot displays the PHP MyAdmin interface. On the left is a navigation tree with a search bar and 'Recent'/'Favorites' tabs. The main area shows the 'ideas' table with a toolbar (Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations) and a status bar indicating 6 rows are shown. A SQL query 'SELECT * FROM `ideas`' is entered. Below the query are options for Profiling, Show all, Number of rows (25), Filter rows (Search this table), and Sort by key (None). The table data is as follows:

	id	title	abstract	publish_date	expiry_date
<input type="checkbox"/>	1	Quyosh Panellari uylar uchun	HOZIRGI KUNGA KELIB U"ZBEKISTANDa ham tabiy energi...	2023-06-10 16:11:00	2023-10-31 16:12:00
<input type="checkbox"/>	2	IT	For frontend	2023-06-13 16:50:00	2023-06-21 16:51:00
<input type="checkbox"/>	3	Bakery	Delicious and famous Bakery house	2023-06-20 14:25:00	2023-08-15 14:25:00
<input type="checkbox"/>	5	Korean Chikken	korean chikken kitchen	2023-06-20 14:27:00	2023-07-19 14:28:00
<input type="checkbox"/>	7	Solar Panel	Solar panel for home	2023-06-20 14:32:00	2023-06-30 14:32:00
<input type="checkbox"/>	8	adjcwkvejcwekjvwrej	fjdvkejrvberkjberkbektjenkjenrk	2023-06-20 16:21:00	2023-07-07 16:21:00

At the bottom, there are 'Check all' and 'With selected:' options with Edit, Copy, Delete, and Export icons, followed by another 'Show all', 'Number of rows' (25), 'Filter rows' (Search this table), and 'Sort by key' (None) section. A Console tab is visible at the very bottom.

PHP MyAdmin은
Admin만 사용 가능

Backend 구성(소스 코드)

```
1 <?php
2
3 namespace Tests\Unit\Api;
4
5 use App\Models\User;
6 use Illuminate\Foundation\Testing\RefreshDatabase;
7 use Illuminate\Foundation\Testing\TestCase;
8 use App\Console\Kernel;
9
10 class LoginTest extends TestCase
11 {
12     use RefreshDatabase;
13
14     /**
15      * Creates the application.
16      *
17      * @return \Illuminate\Foundation\Application
18      */
19     public function createApplication()
20     {
21         $app = require __DIR__ . '/../bootstrap/app.php';
22
23         $app->make(Kernel::class)->bootstrap();
24
25         return $app;
26     }
27
28     public function test_login_success()
29     {
30         // Create a sample user
31         $user = User::factory()->create([
32             'email' => 'test@example.com',
33             'password' => bcrypt('password123'),
34         ]);
35
36         // Send a POST request to the /api/login endpoint
37         $response = $this->postJson('/api/login', [
38             'email' => 'test@example.com',
39             'password' => 'password123',
40         ]);
41
42         // Check if the response has a 200 status code
43         $response->assertStatus(200);
44     }
45 }
```

Login

```
1 <?php
2
3 use Illuminate\Foundation\Inspiring;
4 use Illuminate\Support\Facades\Artisan;
5
6 /*
7 |-----|
8 | Console Routes
9 |-----|
10 |
11 | This file is where you may define all of your Closure based console
12 | commands. Each Closure is bound to a command instance allowing a
13 | simple approach to interacting with each command's IO methods.
14 |
15 */
16
17 Artisan::command('inspire', function () {
18     $this->comment(Inspiring::quote());
19 })->purpose('Display an inspiring quote');
```

Console



Backend 구성(소스 코드)

User

```
1  <?php
2
3  namespace App\Models;
4
5  // use Illuminate\Contracts\Auth\MustVerifyEmail;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7  use Illuminate\Foundation\Auth\User as Authenticatable;
8  use Illuminate\Notifications\Notifiable;
9  use Laravel\Sanctum\HasApiTokens;
10
11 class User extends Authenticatable
12 {
13     use HasApiTokens, HasFactory, Notifiable;
14
15     /**
16      * The attributes that are mass assignable.
17      *
18      * @var array<int, string>
19      */
20     protected $fillable = [
21         'firstname',
22         'lastname',
23         'telephone',
24         'email',
25         'password',
26     ];
27
28     /**
29      * The attributes that should be hidden for serialization.
30      *
31      * @var array<int, string>
32      */
33     protected $hidden = [
34         'password',
35         'remember_token',
36     ];
```

```
37
38     /**
39      * The attributes that should be cast.
40      *
41      * @var array<string, string>
42      */
43     protected $casts = [
44         'email_verified_at' => 'datetime',
45         'product_types' => 'array',
46         'currencies' => 'array',
47         'countries' => 'array',
48     ];
49
50     public function ideas()
51     {
52         return $this->hasMany(Idea::class);
53     }
54
55     public function holdings()
56     {
57         return $this->belongsToMany(Idea::class, 'idea_holders');
58     }
59
60     public function possible_ideas()
61     {
62         return $this->belongsToMany(Idea::class, 'idea_possible_clients');
63     }
64
65     public function rm_recommended_ideas()
66     {
67         return $this->belongsToMany(Idea::class, 'idea_rms');
68     }
69 }
70
```



Backend 구성(소스 코드)

Idea

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Idea extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = ['title', 'abstract', 'publish_date', 'expiry_date', 'content', 'user_id'];
13
14     public function user()
15     {
16         return $this->belongsTo(User::class);
17     }
18
19     public function risk_ratings()
20     {
21         return $this->hasMany(RiskRating::class, 'idea_risk_rating');
22     }
23
24     public function product_types()
25     {
26         return $this->hasMany(ProductType::class, 'idea_product_type');
27     }
28
29     public function major_sectors()
30     {
31         return $this->hasMany(MajorSector::class, 'idea_major_sectors');
32     }
33
34     public function minor_sectors()
35     {
36         return $this->hasMany(MinorSector::class, 'idea_minor_sectors');
37     }
38 }
```

```
39     public function instruments()
40     {
41         return $this->hasMany(Instrument::class, 'idea_instruments');
42     }
43
44     public function currencies()
45     {
46         return $this->hasMany(Currency::class, 'idea_currencies');
47     }
48
49     public function regions()
50     {
51         return $this->hasMany(Region::class, 'idea_regions');
52     }
53
54     public function countries()
55     {
56         return $this->hasMany(Country::class, 'idea_countries');
57     }
58
59     public function holders()
60     {
61         return $this->hasMany(User::class, 'idea_holders');
62     }
63
64     public function possible_clients()
65     {
66         return $this->hasMany(User::class, 'idea_possible_clients');
67     }
68
69     public function rms()
70     {
71         return $this->hasMany(User::class, 'idea_rms');
72     }
73 }
```



Backend 구성(소스 코드)

```
1 <?php
2
3 namespace Tests\Unit\Api;
4
5 use App\Models\User;
6 use Illuminate\Foundation\Testing\RefreshDatabase;
7 use Illuminate\Foundation\Testing\TestCase;
8 use App\Console\Kernel;
9
10 class LoginTest extends TestCase
11 {
12     use RefreshDatabase;
13
14     /**
15      * Creates the application.
16      *
17      * @return \Illuminate\Foundation\Application
18      */
19     public function createApplication()
20     {
21         $app = require __DIR__ . '/../../bootstrap/app.php';
22         $app->make(Kernel::class)->bootstrap();
23         return $app;
24     }
25
26     public function test_login_success()
27     {
28         // Create a sample user
29         $user = User::factory()->create([
30             'email' => 'test@example.com',
31             'password' => bcrypt('password123'),
32         ]);
33
34         // Send a POST request to the /api/login endpoint
35         $response = $this->postJson('/api/login', [
36             'email' => 'test@example.com',
37             'password' => 'password123',
38         ]);
39
40         // Check if the response has a 200 status code
41         $response->assertStatus(200);
42     }
43 }
44
45 }
```

```
2023-06-26 11:45:09 .....
2023-06-26 11:45:09 .....
2023-06-26 11:45:16 .....
2023-06-26 11:45:16 .....
2023-06-26 11:45:17 .....
PS C:\Users\user\Documents\investApp\invest-app\backend> php artisan serve
```

백엔드를 시작하기 위해 php artisan serve 명령이 터미널에 실행됩니다.

```
Warning: PowerShell detected that you might be using a screen reader and has disabled
mpport-Module PSReadLine'.
```

```
PS C:\Users\user\Documents\investApp\invest-app\frontend> npm start
Compiled successfully!
```

프론트엔드를 시작하기 위해 npm start 명령이 터미널에 실행됩니다



결론 및 기대효과

백엔드와 프론트엔드를 사용하여 올바른 장소에서 올바른 아이디어로 구매, 판매 또는 파트너 관계를 맺을 수 있는 웹사이트를 개발했습니다. 이 모든 것은 우리 사이트에서 보호됩니다. 우리는 이 사이트를 더욱 발전시키기 위해 AI를 사용할 계획입니다. 여기서 우리가 얻을 수 있는 가장 큰 도움은 AI가 아이디어를 분류하여 우리에게 최고의 아이디어와 위험을 감수할 수 있는 최고의 아이디어를 제공하는 것입니다. 사이트를 직접 보시고 판단해 보세요.

감사합니다

Invest Idea App