

Project 1 :

Multiple authentication using json :

After I made the authentication for User model that laravel made it ,

First of all I make my model called "Company" , When I sign up by name and bassword that exists in User model I notic that the user is already exist , I notic that the Auth::user() return the user from User model , to solve the problem after hours of searching :

1 – in the auth.php : the guard used by defult is a guard called "api " this name is the middleware that you use in the Route or in the __constucter method in your controller

Ex:

Middleware(auth:api,\$options)

So you must define your custom guard in auth.php ,the provider for your guard will be your table name

Ex:

```
Guards[

'apiCompany' => [
    'driver' => 'jwt',
    'provider' => 'companies',
],
]
```

The 'api' guard is the default guard when you use Auth:: façade

So to spicify your guard you must use auth() helper function

2- back to the api.php you must implement your provider in provider array :

```
'providers' => [
    'users' => [
        'driver' => 'eloquent',
```

```

        'model' => App\Models\User::class,
    ],

    'companies' => [
        'driver' => 'eloquent',
        'model' => App\Models\Company::class,
    ],
],

```

You use `'driver' => 'eloquent',` for models

And `'driver' => 'database',` for tables without models .

SQLSTATE[HY000]: General error: 1005 Can't create table 'ts.#sql-5769_62' (errno: 150)

I search about this error and I tried this to solve it :

- 1- The error caused because the foreign key type and the key type mismatches Like : big integer and integer
- 2- The foreign key must be unsigned if the key is autoIncrement

To solve 1 and 2 I use the `$table->foreignId()` method which automatically solve this problem

3 the error still showed so i tried to change the databases.php in the config folder : i change

`'engine' => null`
To

```
'engine' => 'InnoDB',
```

So that sql don't make some tables by engine and another tables by another engine

But still don't work 😊

4 I doubt in the timestamp method or in the migrate working related with date so the solve can be made in three ways :

1 change the date in the migration file name so the migrate ran from the older table which has the id for the foreign key then go to the table which has it , that's what I do :-)

② Or you can create a new migration for users table instead and insert the course_id

```
php artisan make:migration add_course_id_to_users_table --table=users
```

You then need to use the Schema::table() method (as you're accessing an existing table, not creating a new one). And you can add a column like this:

```
public function up()
{
    Schema::table('users', function($table) {
        $table->integer('course_id');
    });
}
```

and don't forget to add the rollback option:

```
public function down()
{
    Schema::table('users', function($table) {
        $table->dropColumn('course_id');
    });
}
```

Then you can run your migrations:

```
php artisan migrate
```

the image intervention :

you must run composer require intervention/image

I do that then I triggered with an error said "provider intervention/image is not defined " when I add the class to the service provider or alias to the alias array inside the app.php

Some people said it's a composer problem so I reinstall the composer but nothing happened

Some said that I don't install the package in the right way { I reinstall it and run

```
$ php artisan config:cache
```

\$composer dump-autoload

And then run \$composer update

I try to inject the service to mt service provider using

```
$php artisan vendor:publish --provider="Intervention\Image\ImageServiceProviderLaravel5"
```

```
}
```

I try to include the package inn the file without add it in the service provider

It works one time then it crush

I try to include the package with version inside the composer .json requier array 100 times
put it dosent work

At the eng I saw that answer inside laracast website
that says :

Put this inside requier object inside composer.json

```
"intervention/image": "^2.3",
```

Then run the update comand

And it works with that simple 😊

To return a photo from my api after I save it in Storage/uploade dir :

```
storage_path('\uploads\\')
```

return the path to the storage folder then I take the name of my image from the database

```
$headers = ['Content-Type' => 'image/png'];
```

I add the header to my response file

```
return response()->file($path.$name_from_DB , $headers);
```

and I don't use the response method that exist inside the static intervention class Image