

## Assignment -20

First I took a migration call create\_users inside database and Model call User :

Screenshots :



```
public function up(): void
{
    Schema::create('users', function (Blueprint $table) {
        $table->id();
        $table->string('firstName', 50);
        $table->string('lastName', 50);
        $table->string('email', 50)->unique();
        $table->string('mobile', 50);
        $table->string('password', 50);
        $table->string('otp', 50);
        $table->timestamp('created_at')->useCurrent();

        $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate(
    );
    });
}

/**
 * Reverse the migrations.
 */
public function down(): void
{
    Schema::dropIfExists('users');
}
```

In the next page there is a screenshot wher I created a Model. Inside this model I use \$fillable for maintain database table name users.

## Model Screenshot :



Now, I make a migration using command “php artisan migration”

Here is my database screenshot :

| # | Name       | Type        | Collation          | Attributes | Null | Default             | Comments | Extra                         | Action                                      |
|---|------------|-------------|--------------------|------------|------|---------------------|----------|-------------------------------|---|
| 1 | id         | bigint(20)  |                    | UNSIGNED   | No   | None                |          | AUTO_INCREMENT                | <a href="#">Change</a> <a href="#">Drop</a> |
| 2 | firstName  | varchar(50) | utf8mb4_unicode_ci |            | No   | None                |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 3 | lastName   | varchar(50) | utf8mb4_unicode_ci |            | No   | None                |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 4 | email      | varchar(50) | utf8mb4_unicode_ci |            | No   | None                |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 5 | mobile     | varchar(50) | utf8mb4_unicode_ci |            | No   | None                |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 6 | password   | varchar(50) | utf8mb4_unicode_ci |            | No   | None                |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 7 | otp        | varchar(50) | utf8mb4_unicode_ci |            | No   | None                |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 8 | created_at | timestamp   |                    |            | No   | current_timestamp() |          |                               | <a href="#">Change</a> <a href="#">Drop</a> |
| 9 | updated_at | timestamp   |                    |            | No   | current_timestamp() |          | ON UPDATE CURRENT_TIMESTAMP() | <a href="#">Change</a> <a href="#">Drop</a> |

Registration with Exception Handling by Try Catch inside UserController :

Here is a Screenshot :

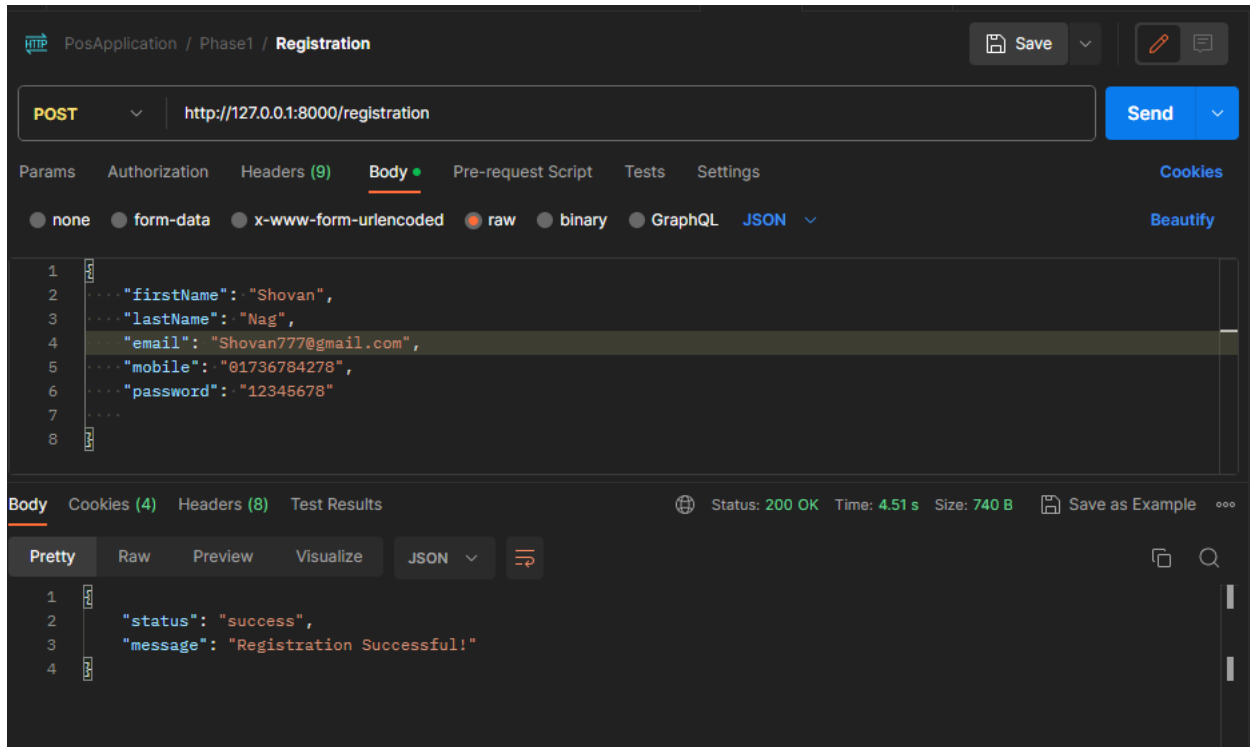
Route:

```
Route::post('/registration',[UserController::class, 'UserRegistration']);
```

UserController :

```
public function UserRegistration(Request $request){  
    try{  
        User::create([  
            'firstName' => $request-> input('firstName'),  
            'lastName' => $request-> input('lastName'),  
            'email' => $request-> input('email'),  
            'mobile' => $request-> input('mobile'),  
            'password' => $request-> input('password'),  
            'otp' => $request-> input('otp')  
        ]);  
        return response()->json([  
            "status"=>"success",  
            "message"=>"Registration Successful!"  
        ]);  
    }catch(Exception $e){  
        return response()->json([  
            "status"=>"failed",  
            "message"=>"Registration failed!"  
        ]);  
    }  
}
```

Here is my postman output for registration :



Here is my database :

Server: 127.0.0.1 » Database: posapplication » Table: users

Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)

`SELECT * FROM `users``

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

|   | id | firstName | lastName | email               | mobile      | password | otp | created_at          | updated_at          |
|---|----|-----------|----------|---------------------|-------------|----------|-----|---------------------|---------------------|
| <input type="checkbox"/> Edit Copy Delete | 1  | Shovan    | Nag      | Shovan@gmail.com    | 01736784278 | 12345678 | 0   | 2023-07-16 12:13:22 | 2023-07-16 12:13:22 |
| <input type="checkbox"/> Edit Copy Delete | 3  | Shovan    | Nag      | Shovan222@gmail.com | 01736784278 | 12345678 | 0   | 2023-07-16 13:02:59 | 2023-07-16 13:02:59 |
| <input type="checkbox"/> Edit Copy Delete | 5  | Shovan    | Nag      | Shovan777@gmail.com | 01736784278 | 12345678 | 0   | 2023-07-18 14:35:24 | 2023-07-18 14:35:24 |

Check all | With selected: Edit Copy Delete Export

From the next page I'll show JWT For Authentication. To be continue..

Here is a Screenshot for JWT Authentication :

1<sup>st</sup> I created a command “composer require firebase/php-jwt”. And created a JWT\_TOKEN inside .env file



```
JWT_TOKEN = 123SGHTYVBKLBKHSDFsdfsfg5621
```



```
<?php
namespace App\Helper;
use Exception;
use Firebase\JWT\JWT;
use Firebase\JWT\Key;

class JWTToken
{
    public static function CreateToken($userEmail):string{
        $key = env('JWT_TOKEN');
        $payload =[
            'iss'=>'Laravel Token',
            'iat'=> time(),
            'exp'=>time()+60*60,
            'userEmail'=>$userEmail
        ];

        return JWT::encode($payload, $key, 'HS256');
    }

    public static function VerifyToken($token):string{
        try{
            $key = env('JWT_TOKEN');
            $decode = JWT::decode($token, new Key($key, 'HS256'));
            return $decode->userEmail;

        }catch(Exception $e){
            return 'Unothorized';
        }
    }
}
```

Now, I have created User Login and issued JWT Token with registered user.

Here is all screenshots below..

Route :



When I hit this URL then output by postman is like :

