

The application has two parts

1. Worker
2. Consumer

WORKER Installation

1. Create twidpay project by the following command

```
Composer create-project --prefer-dist laravel/lumen twidpay
```

2. Since no Artisan is available by default in Laravel Lumen, hence, we have to install the Artisan.

I have used [flipboxstudio/lumen-generator](https://github.com/flipboxstudio/lumen-generator) package

Use the below code to install

```
Composer require laravel/lumen-generator
```

Inside bootstrap/app.php file add the following snippet:

```
$app->register(Flipbox\LumenGenerator\LumenGeneratorServiceProvider::class);
```

3. Activate Eloquent and Facades uncommenting
app->withEloquent
app->withFacades
4. Create a Database twidpay_db and mention in the .env file
5. Generate Key in .env file

```
php artisan key:generate
```

6. Create migration
Php artisan make:migration create_pincode_table

Add Schema

```
Schema::create('pincode', function (Blueprint $table) {  
    $table->id();  
    $table->string('officename');  
    $table->integer('pincode');  
    $table->string('officeType');  
    $table->string('Deliverystatus');  
    $table->string('divisionname');  
    $table->string('regionname');
```

```

        $table->string('circlename');
        $table->string('Taluk');
        $table->string('Districtname');
        $table->string('statename');
        $table->timestamps();

    });

```

7. Create PincodeController with resource(If CRUD needed) & Model
php artisan make:controller PincodeController -r -m Pincode

8. Create storePincode function for storing the data

```

public function storePincode($delimiter=','){

    $filename =
'http://data.gov.in/sites/default/files/all_india_pin_code.csv';

    $file = fopen ( $filename, 'r' );

    if (($handle = $file) !== FALSE) {
        $count = 0;
        while ( ($row = fgetcsv ( $handle, 1000, $delimiter ))
        !== FALSE ) {

            $count++;
            if ($count > 1) {
                $csv_data = new Pincode ();
                $csv_data->officename = $row[0];
                $csv_data->pincode = $row[1];
                $csv_data->officeType = $row[2];
                $csv_data->Deliverystatus = $row[3];
                $csv_data->divisionname = $row[4];
                $csv_data->regionname = $row[5];
                $csv_data->circlename = $row[6];
                $csv_data->Taluk = $row[7];
                $csv_data->Districtname = $row[8];
                $csv_data->statename = $row[9];

                $csv_data->save ();
            }
        }
    }
}

```

```

        }
        fclose ($handle);
    }
    return $row;
}

```

9. Set the Routes in API group,

```

$router->group(['prefix'=>'api'], function() use($router){

    $router->get('pincodes','PincodeController@index');
    $router->get('store-pincode','PincodeController@storePincode');

});

```

Installation of Consumer:

1. Install laravel by using composer command
Composer create-project --prefer-dist laravel/laravel consumer
2. Create MainController and mention the below code also mention paginate function there:

Php artisan make:controller MainController

```

<?php

namespace App\Http\Controllers;

use App\Models\Main;
use Illuminate\Http\Request;
use Illuminate\Pagination\Paginator;
use Illuminate\Support\Collection;
use Illuminate\Pagination\LengthAwarePaginator;

class MainController extends Controller
{
    /**
     * Display a listing of the resource.
     */
}

```

```

    *
    * @return \Illuminate\Http\Response
    */
    public function index()
    {
        $data =
        json_decode(file_get_contents('http://127.0.0.1:8001/api/pincodes
        '), true);

        $myCollectionObj = collect($data);

        $pincodes = $this->paginate($myCollectionObj);

        return view('welcome',compact('pincodes'));
    }

    /**
     * The attributes that are mass assignable.
     *
     * @var array
     */
    public function paginate($items, $perPage = 5, $page = null,
    $options = [])
    {
        $page = $page ?: (Paginator::resolveCurrentPage() ?: 1);
        $items = $items instanceof Collection ? $items :
        Collection::make($items);
        return new LengthAwarePaginator($items->forPage($page,
        $perPage), $items->count(), $perPage, $page);
    }

```

3. Create Blade file as mentioned below:

```

<!DOCTYPE html>
<html>

<head>
    <meta charset="utf-8" />

```

```

        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <title>Laravel Pagination Demo</title>
        <meta name="viewport" content="width=device-width,
initial-scale=1">
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/boot
strap.min.css">

        <style>
            .shadow-sm{display:none !important;}
        </style>
    </head>

    <body>
        <div class="container mt-5">
            <table class="table table-bordered mb-5">
                <thead>
                    <tr class="table-success">
                        <th scope="col">#</th>
                        <th scope="col">Office Name</th>
                        <th scope="col">Pincode</th>
                        <th scope="col">Office type</th>
                        <th scope="col">Delivery status</th>
                        <th scope="col">Division name</th>
                        <th scope="col">Region name</th>
                        <th scope="col">Circle name</th>
                        <th scope="col">Taluk</th>
                        <th scope="col">District name</th>
                        <th scope="col">State name</th>
                    </tr>
                </thead>
                <tbody>
                    @foreach($pincodes as $pincode)
                        <tr>
                            <th scope="row">{{ $pincode['id'] }}</th>
                            <td>{{ $pincode['officename'] }}</td>
                            <td>{{ $pincode['pincode'] }}</td>
                            <td>{{ $pincode['officeType'] }}</td>
                            <td>{{ $pincode['Deliverystatus'] }}</td>
                            <td>{{ $pincode['divisionname'] }}</td>
                            <td>{{ $pincode['regionname'] }}</td>
                            <td>{{ $pincode['circlename'] }}</td>
                            <td>{{ $pincode['Taluk'] }}</td>

```

```

                <td>{{ $pincode['Districtname'] }}</td>
                <td>{{ $pincode['statename'] }}</td>
            </tr>
        @endforeach
    </tbody>
</table>
    {{ $pincodes->links() }}

</div>
</body>
</html>

```

4. Create route :

```

<?php

use Illuminate\Support\Facades\Route;

/*
|-----
|
| Web Routes
|-----
|
| Here is where you can register web routes for your application.
These
| routes are loaded by the RouteServiceProvider within a group
which
| contains the "web" middleware group. Now create something
great!
|
*/

Route::get('/', 'App\Http\Controllers\MainController@index');

```

