HireGallaxy Documentation

Content

1. Website Overview :

★ Welcome to Hiregallaxy Website documentation. This website built to provide job opportunity to the employers and the job seekers. Employers can easily register and post their job to this website. Then Job seekers or candidates can apply that job by uploading their resume. Employers can register, login, post job, manage their jobs, see applications, contact with job candidates through messaging and many more. Candidates can register, login, change profile like – add education, skills, portfolio, resume, abouts and everything, apply job, modify their application after applying jobs, make job favorite and many more features. And Site admin can Manage jobs, manage categories, manage cities, manage employers, manage candidates, manage job templates, manage job crawlers, manage site settings like logo, favicon and so on.

★ At the same time website has python crawler which crawls prefinded website at predefined cycle every day and scrap the ads data such as title , name , details and location of posted job.

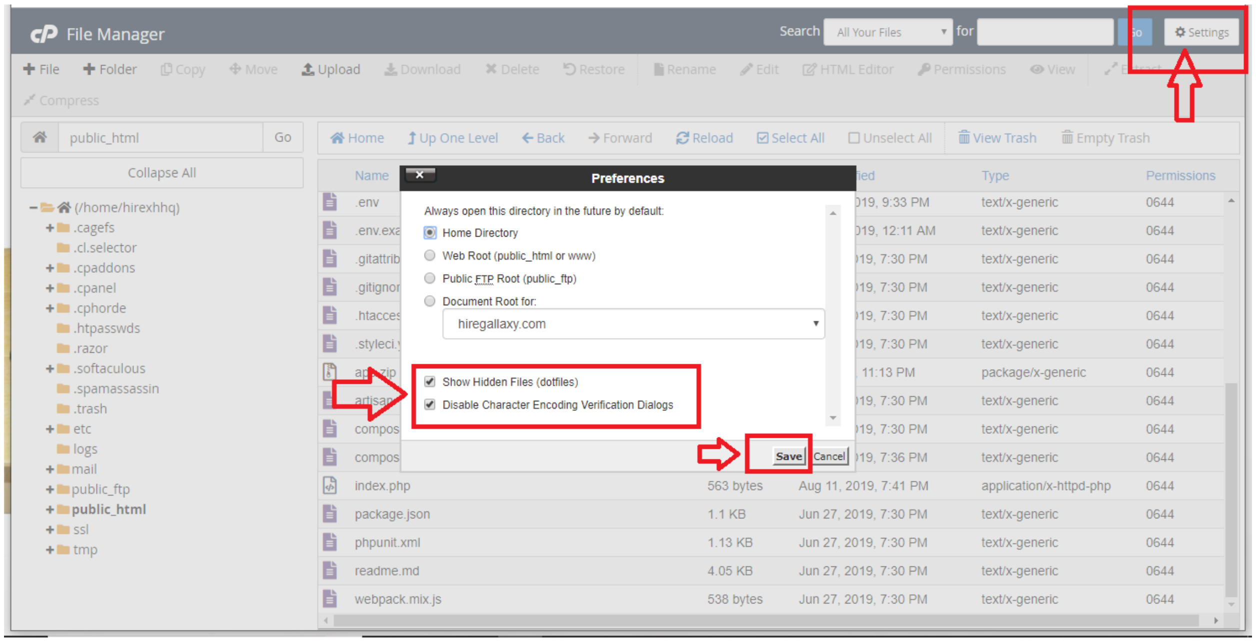
2. Programming Language:

❖ Hiregallaxy website is built with PHP. For frontend framework Vue js is used and as backend Laravel PHP framework is used. All other languages used in the websites are.

- ➔ HTML 5
- ➔ CSS 3
- ➔ Bootstrap 4 (CSS Framework)
- ➔ JavaScript
- ➔ Jquery 3.2
- ➔ Vue JS 2.0 (Frontend JS Framework)
- ➔ PHP 7 (Server Language)
- ➔ Laravel 5.8 (Backend Framework)
- ➔ MySQL (database)

3. Software Setup :

- ➢ Login to cpanel -> File Manager ->
- ➢ -> public_html -> Extract the app.zip
- ➢ To modify database and site settings go to .env file and modify the .env file as your needs.
- ➢ But by default in cpanel, the dot files, like .env, .env.example are not allowed to see.
- ➢ You have to go to Cpanel -> File Manager
- ➢ click the check mark to display .env files.
- ➢ See the screenshot below .



Preferences

Always open this directory in the future by default:

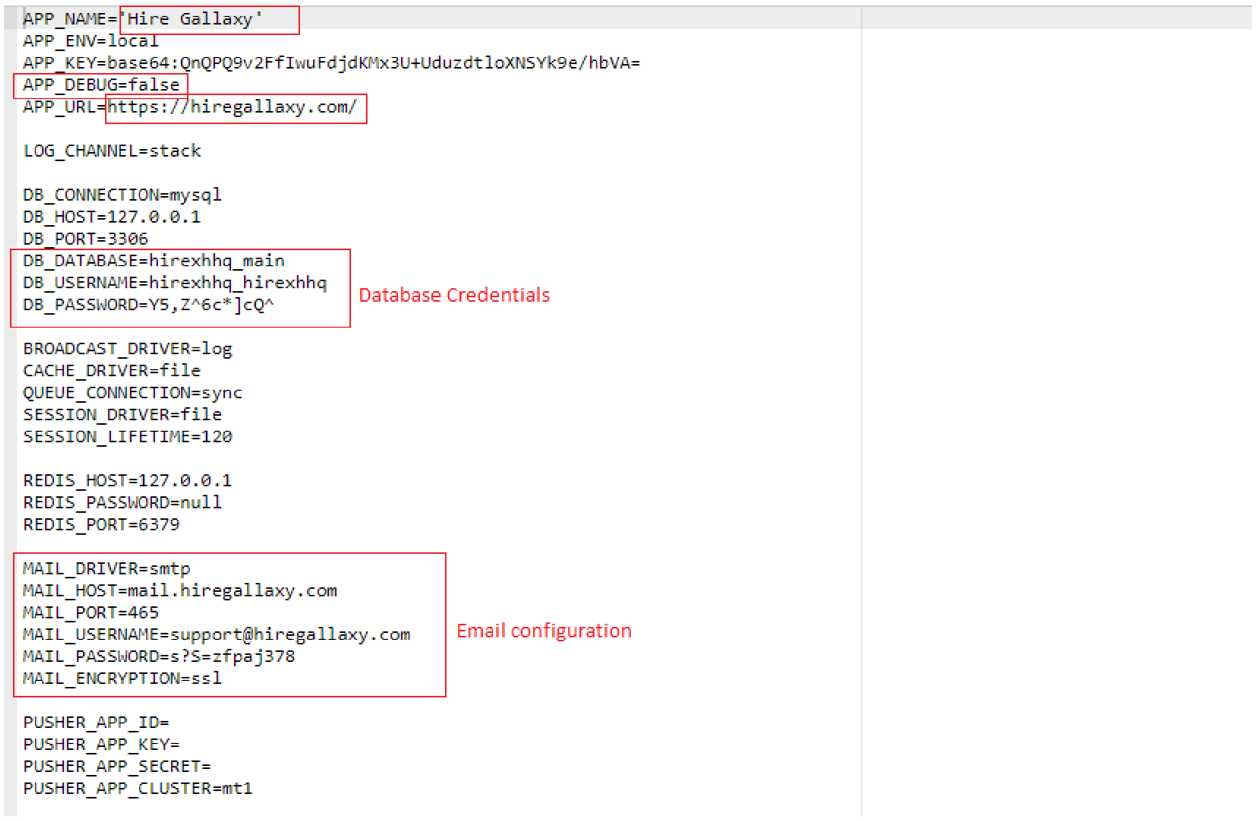
- [✓ Home Directory]
- Web Root (public_html or www)
- Public FTP Root (public_ftp)
- Document Root for: hiregallaxy.com

- [✓] Show Hidden Files (dotfiles)
- [✓] Disable Character Encoding Verification Dialogs

Save Cancel

Name		ied	Type	Permissions
.env		019, 9:33 PM	text/x-generic	0644
.env.exa		019, 12:11 AM	text/x-generic	0644
.gitattrib		19, 7:30 PM	text/x-generic	0644
.gitignor		19, 7:30 PM	text/x-generic	0644
.htacces		19, 7:30 PM	text/x-generic	0644
.styleci.y		19, 7:30 PM	text/x-generic	0644
ap zip		11:13 PM	package/x-generic	0644
artisa		19, 7:30 PM	text/x-generic	0644
compos		19, 7:30 PM	text/x-generic	0644
compos		19, 7:36 PM	text/x-generic	0644
index.php	563 bytes	Aug 11, 2019, 7:41 PM	application/x-httpd-php	0644
package.json	1.1 KB	Jun 27, 2019, 7:30 PM	text/x-generic	0644
phpunit.xml	1.13 KB	Jun 27, 2019, 7:30 PM	text/x-generic	0644
readme.md	4.05 KB	Jun 27, 2019, 7:30 PM	text/x-generic	0644
webpack.mix.js	538 bytes	Jun 27, 2019, 7:30 PM	text/x-generic	0644

➢ See below screenshot to edit now the .env file:



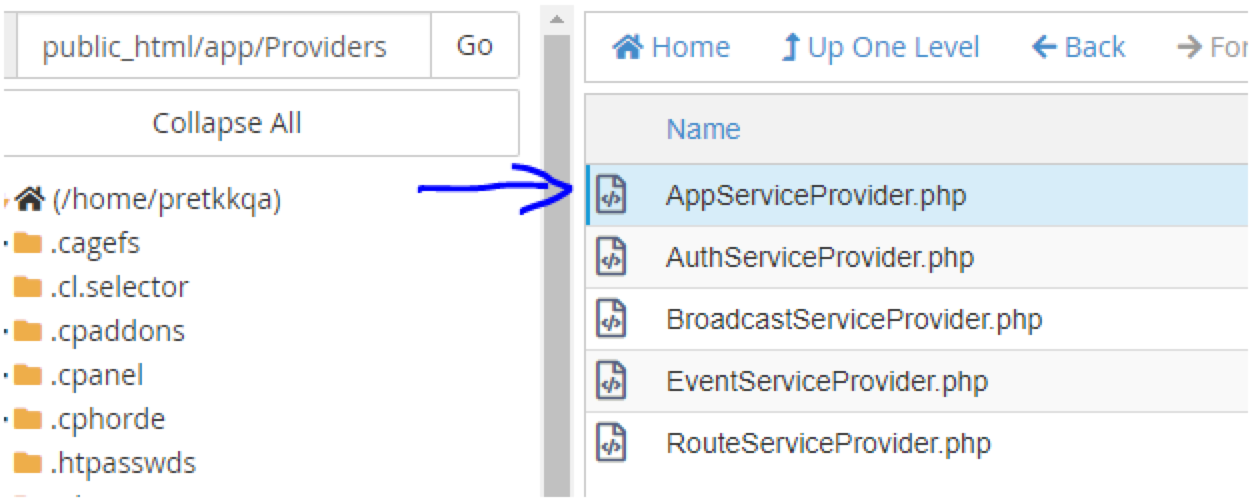
```
APP_NAME='Hire Gallaxy'
APP_ENV=local
APP_KEY=base64:QnQPQ9v2FfIwuFdjdKMx3U+UduzdtloXNSYk9e/hbVA=
APP_DEBUG=false
APP_URL=https://hiregallaxy.com/

LOG_CHANNEL=stack

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=hirexhhq_main
DB_USERNAME=hirexhhq_hirexhhq
DB_PASSWORD=Y5,Z^6c*]cQ^

BROADCAST_DRIVER=log
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_DRIVER=smtp
MAIL_HOST=mail.hiregallaxy.com
MAIL_PORT=465
MAIL_USERNAME=support@hiregallaxy.com
MAIL_PASSWORD=s?S=zfpaj378
MAIL_ENCRYPTION=ssl

PUSHER_APP_ID=
PUSHER_APP_KEY=
PUSHER_APP_SECRET=
PUSHER_APP_CLUSTER=mt1
```

➢ That's it. Your project will be set up. Now it's time to setup the database. Without the database, nothing will give you the exact result.
➢ To view code errors change the above APP_DEBUG=false > APP_DEBUG=true
➢ Otherwise make it to APP_DEBUG=false.

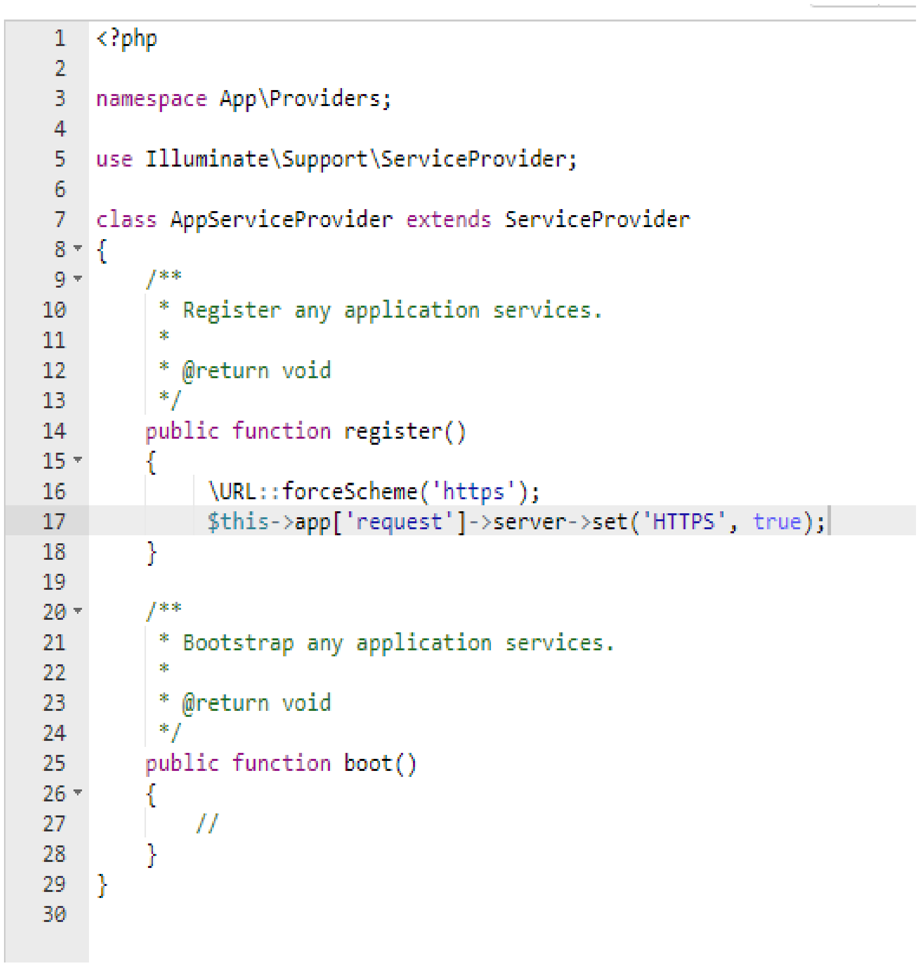
5. Database File Setup :

❖ Create a database by above DB_DATABASE in the .env file. Like, hirexhhq_main.
❖ Then create user by above DB_USERNAME like hirehhq_hirexhq and with a password like above password.

- ❖ Then give the user all permissions. Sometime you may have created the user, but you don't give the user all of the permission to manage database.
- ❖ Now import your hiregallaxy.sql file in the database and run your project.
- ❖ Your project will start at https://hiregallaxy.com.
- ❖ To force your site to https not http, you may go to app/Providers/AppServiceProvider.php



- ❖ And then comment or uncomment the lines inside the register() function. To enable https, then take this remain the same. To run over http, make this in commented (//).

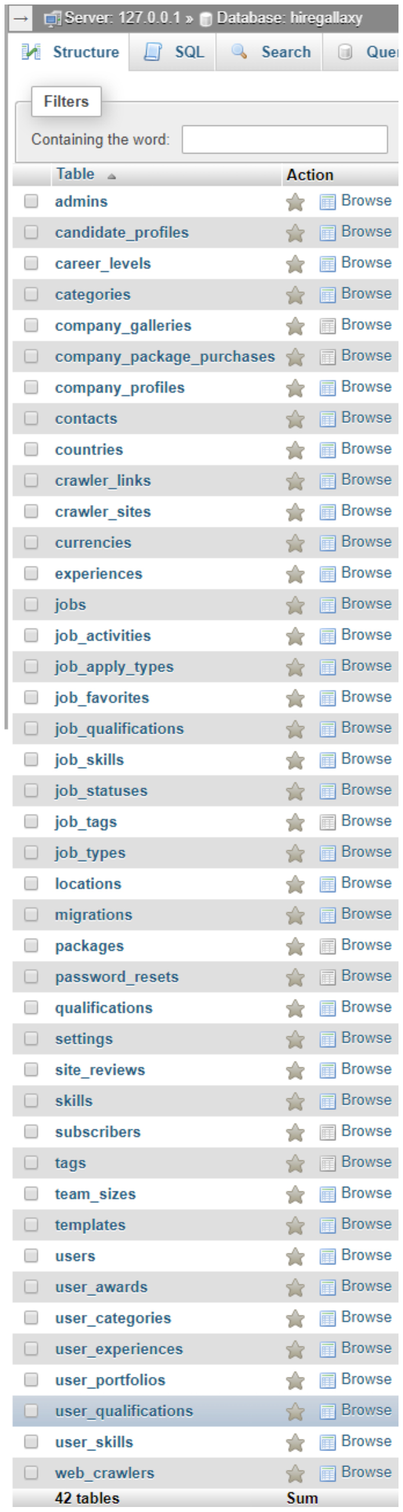


```php
<?php

namespace App\Providers;

use Illuminate\Support\ServiceProvider;

class AppServiceProvider extends ServiceProvider
{
    /**
     * Register any application services.
     *
     * @return void
     */
    public function register()
    {
        \URL::forceScheme('https');
        $this->app['request']->server->set('HTTPS', true);
    }

    /**
     * Bootstrap any application services.
     *
     * @return void
     */
    public function boot()
    {
        //
    }
}

```

5. Database Schema Design :

- ★ Hiregallaxy app is based on mysql database.
- ★ There is a total of 42 tables in the hiregallaxy database.
- ★ The main tables of this sql is: users, company_profiles, candidate_profiles, categories, countries, jobs, job_activities, settings, templates and other tables are basically child tables of the parent tables.
- ★ *Like for users and candidate_profiles table* - it belongs to the following tables: candidate_profiles, user_portfolios, user_awards, user_skills, user_qualifications, user_experiences.

★ *For users and company_profiles table* - it belongs to the following tables: company_profiles, company_galleries, team_sizes.
★ *For jobs table* - it belongs to the following tables: job_activities, job_apply_types, job_favorites, job_qualification, job_skills, job_statuses, job_tags, job_types.

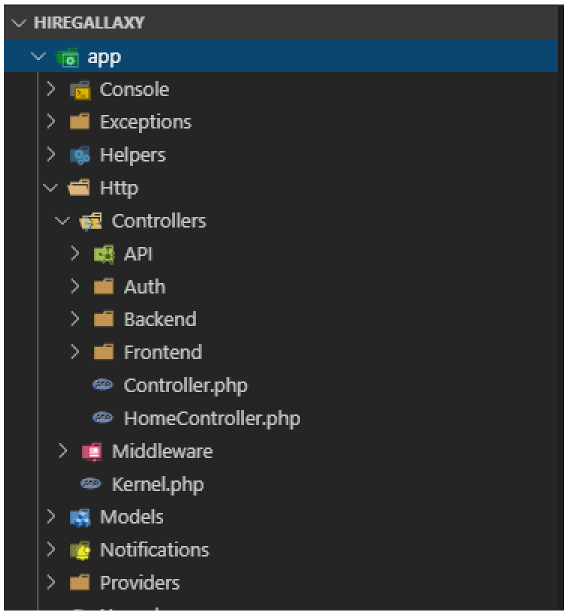


Containing the word:

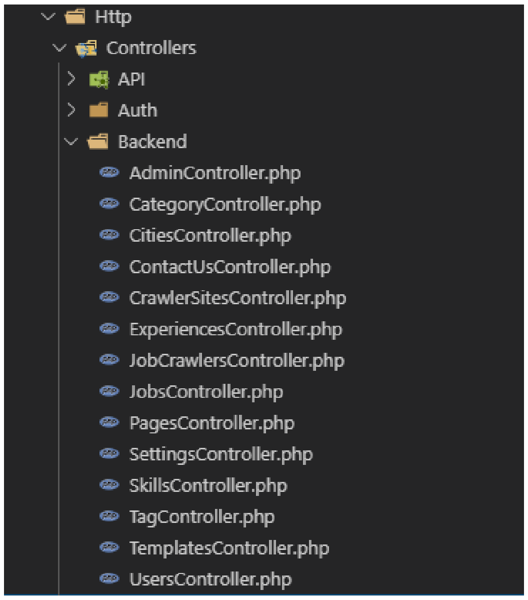
Table	Action
admins	Browse
candidate_profiles	Browse
career_levels	Browse
categories	Browse
company_galleries	Browse
company_package_purchases	Browse
company_profiles	Browse
contacts	Browse
countries	Browse
crawler_links	Browse
crawler_sites	Browse
currencies	Browse
experiences	Browse
jobs	Browse
job_activities	Browse
job_apply_types	Browse
job_favorites	Browse
job_qualifications	Browse
job_skills	Browse
job_statuses	Browse
job_tags	Browse
job_types	Browse
locations	Browse
migrations	Browse
packages	Browse
password_resets	Browse
qualifications	Browse
settings	Browse
site_reviews	Browse
skills	Browse
subscribers	Browse
tags	Browse
team_sizes	Browse
templates	Browse
users	Browse
user_awards	Browse
user_categories	Browse
user_experiences	Browse
user_portfolios	Browse
user_qualifications	Browse
user_skills	Browse
web_crawlers	Browse
42 tables	Sum

5) File Structure:

- ❏ Hiregallaxy app is based on Laravel PHP framework and so it is following the Laravel framework project structure. Here is the full documentation of Laravel: https://laravel.com/docs/5.8.
- ❏ **app folder:** app folder is the heart of a laravel framework. In this folder, there is the models of database file to make eloquent relationships, the controllers, notifications, middleware setup.

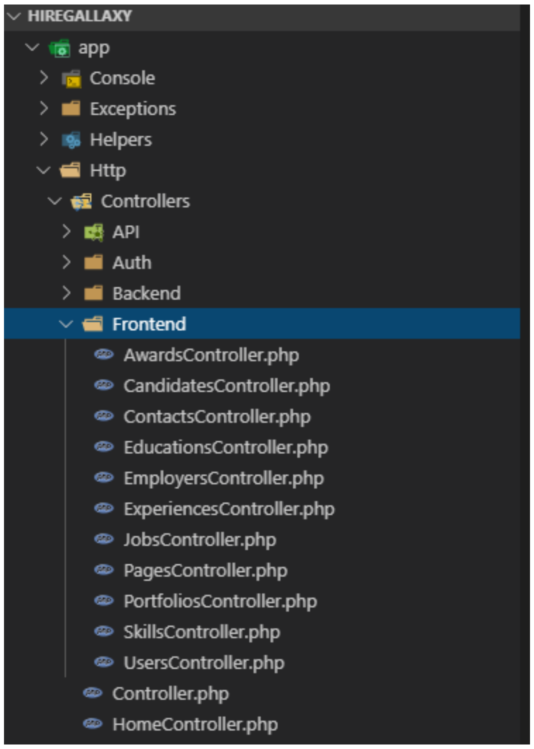


- ❏ **app/Http/Controllers:** In this folder all the controllers are stored by which any action is called. There are two folders also for making frontend controllers and backend controllers.
- ❏ Here is the backend folder controller files:



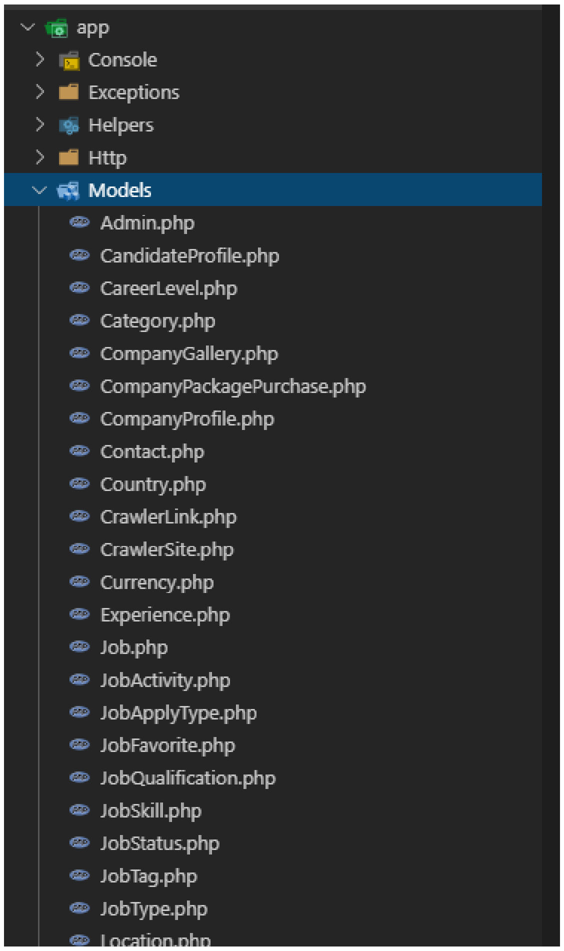
```
Http
  Controllers
    API
    Auth
    Backend
      AdminController.php
      CategoryController.php
      CitiesController.php
      ContactUsController.php
      CrawlerSitesController.php
      ExperiencesController.php
      JobCrawlersController.php
      JobsController.php
      PagesController.php
      SettingsController.php
      SkillsController.php
      TagController.php
      TemplatesController.php
      UsersController.php
```

❑ And here is the frontend folder controller files:



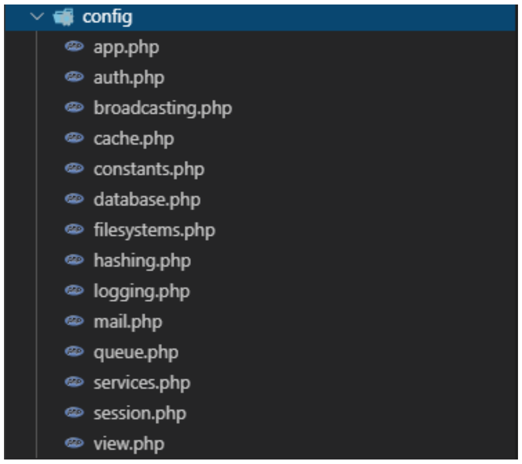
```
HIREGALLAXY
  app
    Console
    Exceptions
    Helpers
    Http
      Controllers
        API
        Auth
        Backend
        Frontend
          AwardsController.php
          CandidatesController.php
          ContactsController.php
          EducationsController.php
          EmployersController.php
          ExperiencesController.php
          JobsController.php
          PagesController.php
          PortfoliosController.php
          SkillsController.php
          UsersController.php
        Controller.php
        HomeController.php
```

❑ **app/Models:** All the Models to make eloquent database queries.



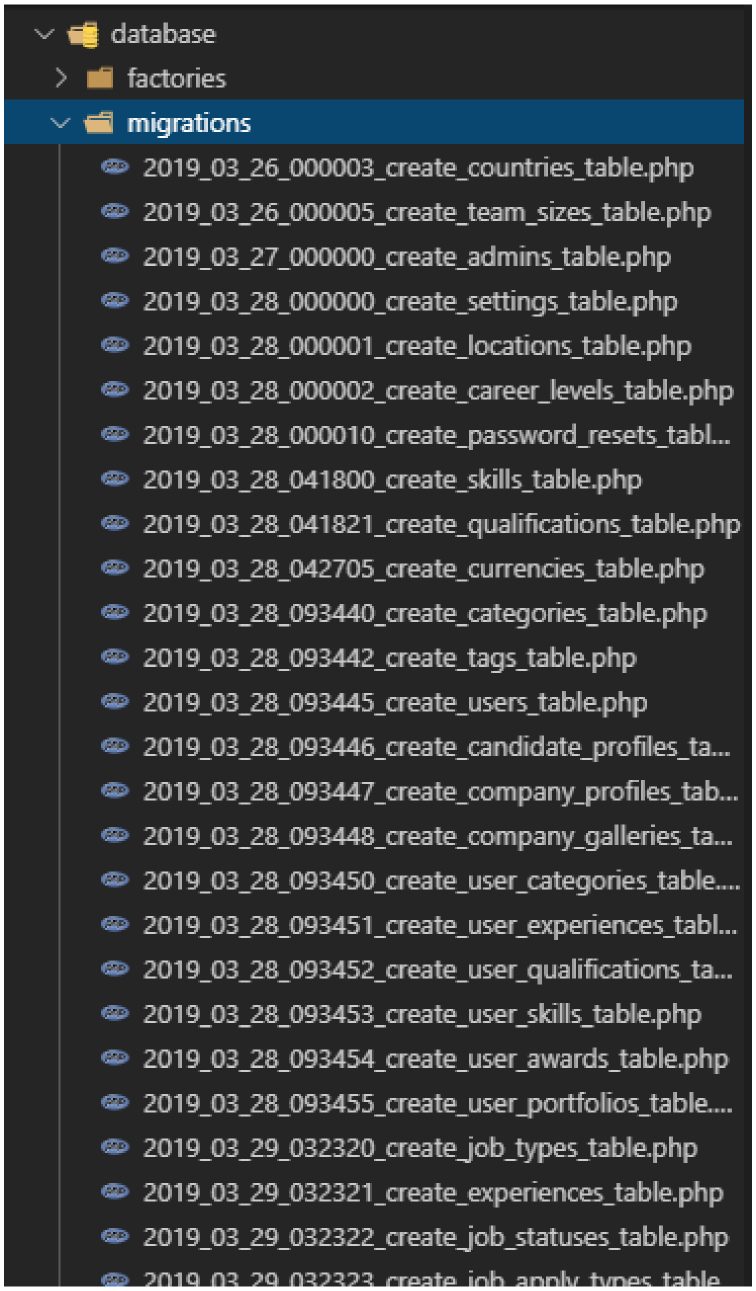
```
app
  Console
  Exceptions
  Helpers
  Http
  Models
    Admin.php
    CandidateProfile.php
    CareerLevel.php
    Category.php
    CompanyGallery.php
    CompanyPackagePurchase.php
    CompanyProfile.php
    Contact.php
    Country.php
    CrawlerLink.php
    CrawlerSite.php
    Currency.php
    Experience.php
    Job.php
    JobActivity.php
    JobApplyType.php
    JobFavorite.php
    JobQualification.php
    JobSkill.php
    JobStatus.php
    JobTag.php
    JobType.php
    Location.php
```

- ❑ **config/:** In this folder of the configuration of the sites are done. This is actually built with Laravel by default. You don't need any customization here.



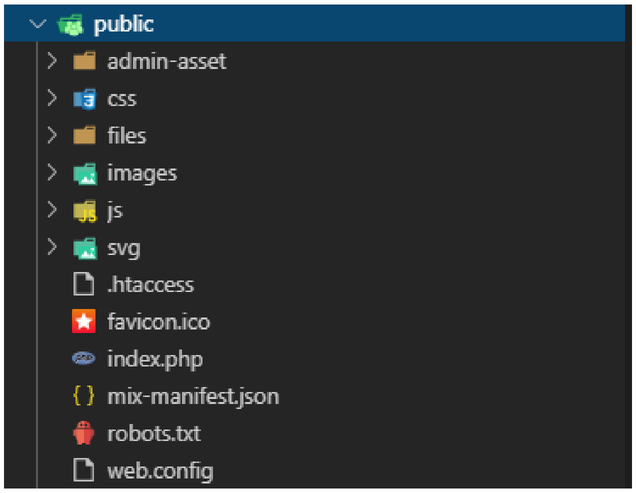
```
config
    app.php
    auth.php
    broadcasting.php
    cache.php
    constants.php
    database.php
    filesystems.php
    hashing.php
    logging.php
    mail.php
    queue.php
    services.php
    session.php
    view.php
```

- ❑ **database/migrations:** In this folder all of the migrations of database tables are done. This is used for project startup database migration.



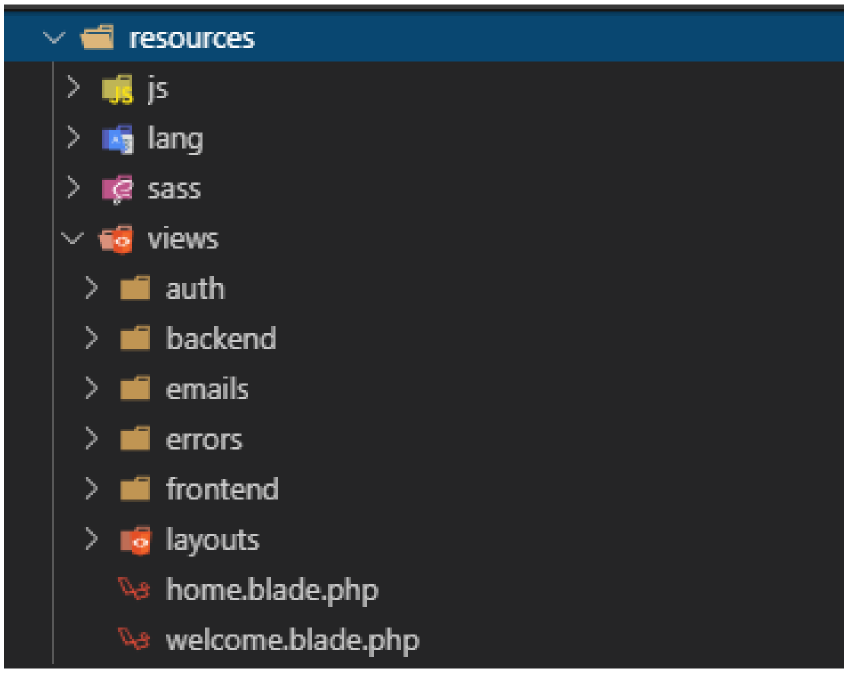
```
database
  factories
  migrations
    2019_03_26_000003_create_countries_table.php
    2019_03_26_000005_create_team_sizes_table.php
    2019_03_27_000000_create_admins_table.php
    2019_03_28_000000_create_settings_table.php
    2019_03_28_000001_create_locations_table.php
    2019_03_28_000002_create_career_levels_table.php
    2019_03_28_000010_create_password_resets_tabl...
    2019_03_28_041800_create_skills_table.php
    2019_03_28_041821_create_qualifications_table.php
    2019_03_28_042705_create_currencies_table.php
    2019_03_28_093440_create_categories_table.php
    2019_03_28_093442_create_tags_table.php
    2019_03_28_093445_create_users_table.php
    2019_03_28_093446_create_candidate_profiles_ta...
    2019_03_28_093447_create_company_profiles_tab...
    2019_03_28_093448_create_company_galleries_ta...
    2019_03_28_093450_create_user_categories_table....
    2019_03_28_093451_create_user_experiences_tabl...
    2019_03_28_093452_create_user_qualifications_ta...
    2019_03_28_093453_create_user_skills_table.php
    2019_03_28_093454_create_user_awards_table.php
    2019_03_28_093455_create_user_portfolios_table....
    2019_03_29_032320_create_job_types_table.php
    2019_03_29_032321_create_experiences_table.php
    2019_03_29_032322_create_job_statuses_table.php
    2019_03_29_032323_create_job_apply_types_table
```

- ❑ **public/:** In this folder all of the assets (css, js, images) of the sites are stored.



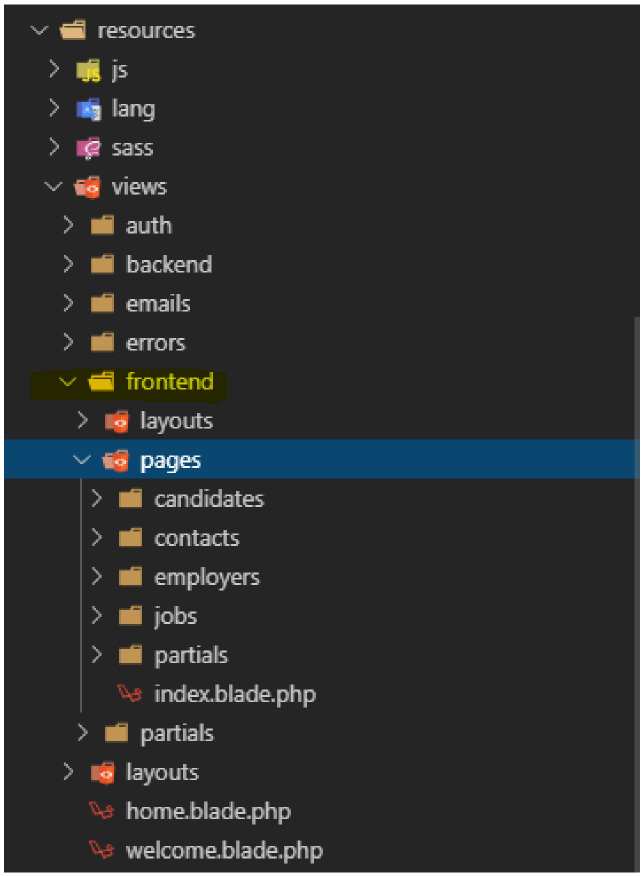
```
public
  admin-asset
  css
  files
  images
  js
  svg
  .htaccess
  favicon.ico
  index.php
  mix-manifest.json
  robots.txt
  web.config
```

- ❑ **resources/views:** In this folder all of the views which is responsible to render in the web pages are done. There are two major folder here also, like frontend and backend to minimize.



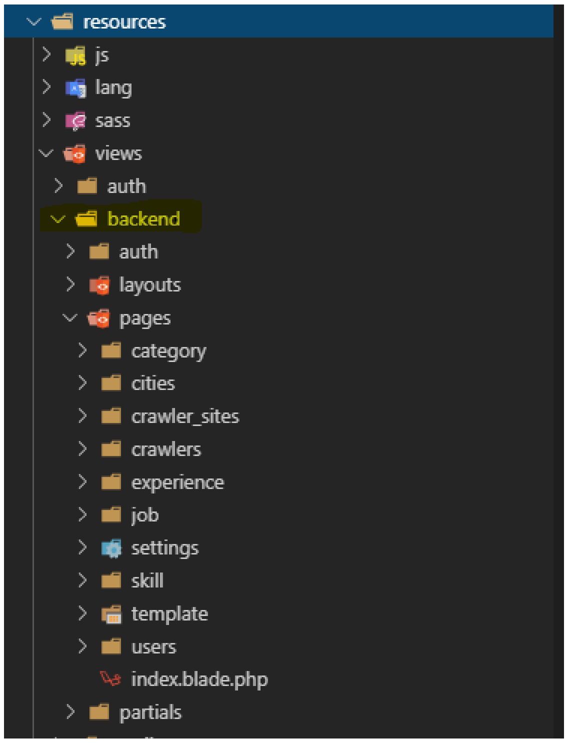
```
resources
  js
  lang
  sass
  views
    auth
    backend
    emails
    errors
    frontend
    layouts
    home.blade.php
    welcome.blade.php
```

- ❑ **<u>resources/views/frontend:</u>** In this folder all the frontend views are stored. Which is responsible for frontend views of the website.



```
resources
  js
  lang
  sass
  views
    auth
    backend
    emails
    errors
    frontend
      layouts
      pages
        candidates
        contacts
        employers
        jobs
        partials
        index.blade.php
      partials
    layouts
    home.blade.php
    welcome.blade.php
```

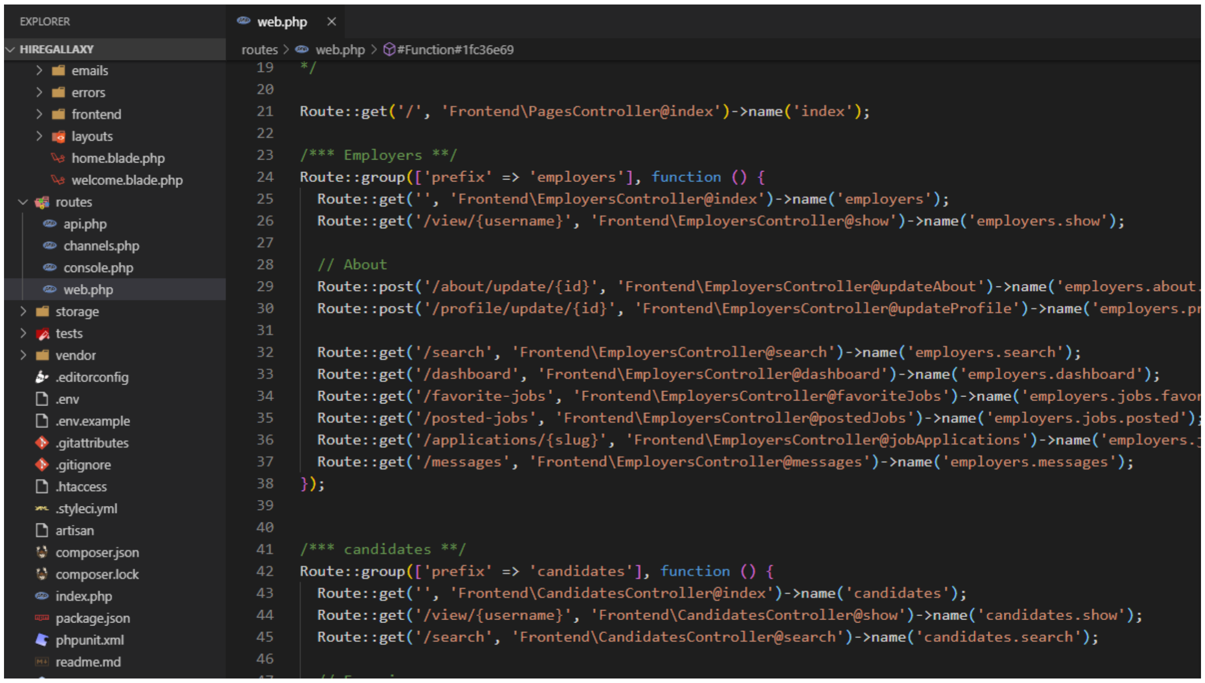
- ❑ **resources/views/backend:** In this folder all the backend views are stored. Which is responsible for backend/admin views of the website.



```
resources
  js
  lang
  sass
  views
    auth
    backend
      auth
      layouts
      pages
        category
        cities
        crawler_sites
        crawlers
        experience
        job
        settings
        skill
        template
        users
        index.blade.php
      partials
```

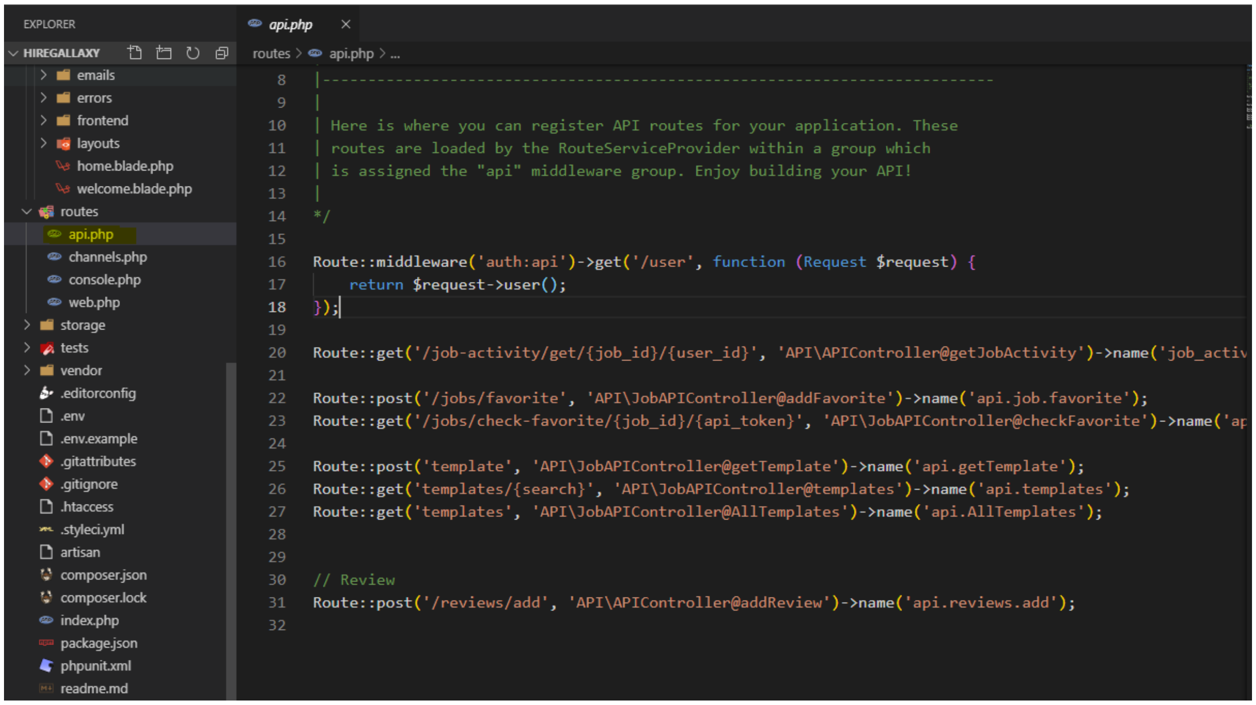
- ❑ **routes/:** In this folder all of the routes or url which is hit by us for the website. Like hiregallaxy.com/jobs, here /jobs is the route and that needs to be ensured from which controller and function will be called for this.
- ❑ There are two actual routes. For Website that is in the web.php and for API data like when we hit ajax, we will use the api.php routes.

- ❑ **routes/web.php:** Website routes



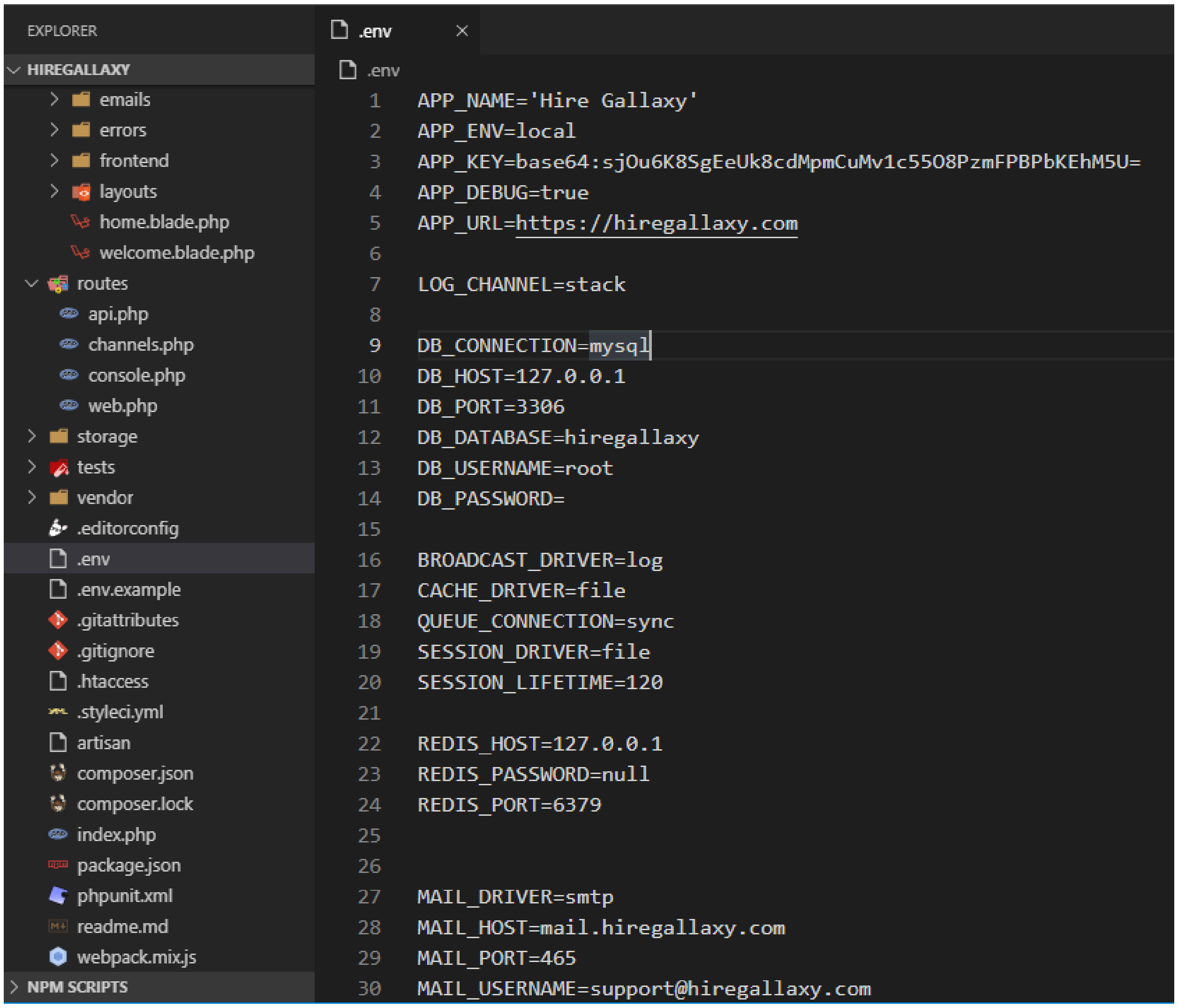
```php
*/

Route::get('/', 'Frontend\PagesController@index')->name('index');

/*** Employers **/
Route::group(['prefix' => 'employers'], function () {
  Route::get('', 'Frontend\EmployersController@index')->name('employers');
  Route::get('/view/{username}', 'Frontend\EmployersController@show')->name('employers.show');

  // About
  Route::post('/about/update/{id}', 'Frontend\EmployersController@updateAbout')->name('employers.about.
  Route::post('/profile/update/{id}', 'Frontend\EmployersController@updateProfile')->name('employers.pr

  Route::get('/search', 'Frontend\EmployersController@search')->name('employers.search');
  Route::get('/dashboard', 'Frontend\EmployersController@dashboard')->name('employers.dashboard');
  Route::get('/favorite-jobs', 'Frontend\EmployersController@favoriteJobs')->name('employers.jobs.favor
  Route::get('/posted-jobs', 'Frontend\EmployersController@postedJobs')->name('employers.jobs.posted');
  Route::get('/applications/{slug}', 'Frontend\EmployersController@jobApplications')->name('employers.j
  Route::get('/messages', 'Frontend\EmployersController@messages')->name('employers.messages');
});


/*** candidates **/
Route::group(['prefix' => 'candidates'], function () {
  Route::get('', 'Frontend\CandidatesController@index')->name('candidates');
  Route::get('/view/{username}', 'Frontend\CandidatesController@show')->name('candidates.show');
  Route::get('/search', 'Frontend\CandidatesController@search')->name('candidates.search');

```

- ❑ **routes/api.php:** API routes



```
|--------------------------------------------------------------------------
|
| Here is where you can register API routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| is assigned the "api" middleware group. Enjoy building your API!
|
*/

Route::middleware('auth:api')->get('/user', function (Request $request) {
    return $request->user();
});

Route::get('/job-activity/get/{job_id}/{user_id}', 'API\APIController@getJobActivity')->name('job_activ

Route::post('/jobs/favorite', 'API\JobAPIController@addFavorite')->name('api.job.favorite');
Route::get('/jobs/check-favorite/{job_id}/{api_token}', 'API\JobAPIController@checkFavorite')->name('ap

Route::post('template', 'API\JobAPIController@getTemplate')->name('api.getTemplate');
Route::get('templates/{search}', 'API\JobAPIController@templates')->name('api.templates');
Route::get('templates', 'API\JobAPIController@AllTemplates')->name('api.AllTemplates');


// Review
Route::post('/reviews/add', 'API\APIController@addReview')->name('api.reviews.add');
```

- ❑ **.env:** All configuration variables and data for our site.

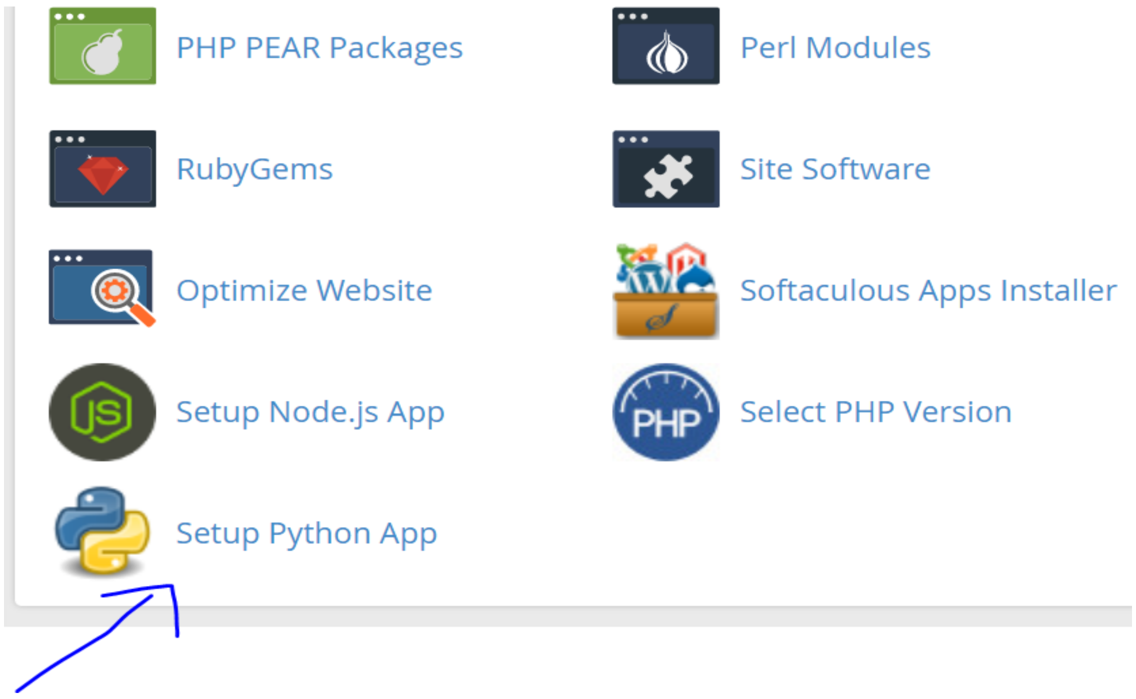


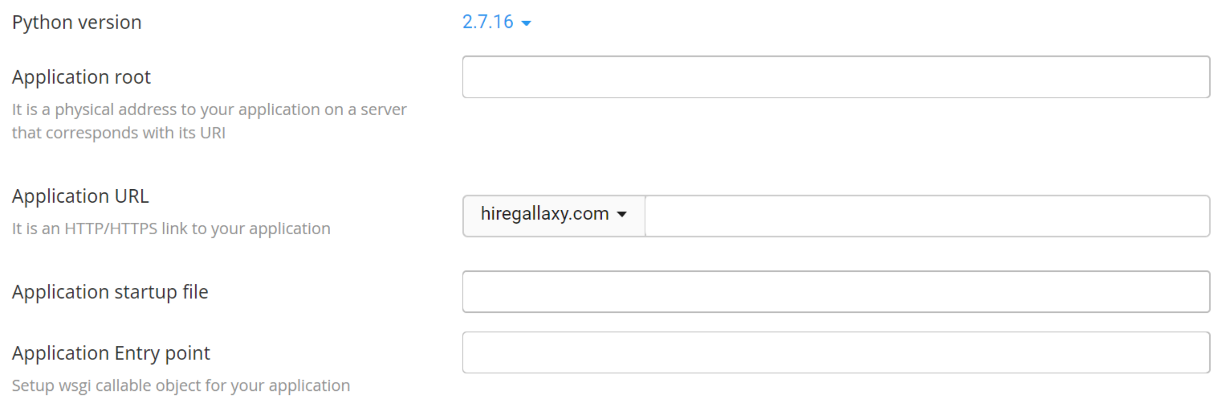
```
APP_NAME='Hire Gallaxy'
APP_ENV=local
APP_KEY=base64:sjOu6K8SgEeUk8cdMpmCuMv1c55O8PzmFPBPbKEhM5U=
APP_DEBUG=true
APP_URL=https://hiregallaxy.com

LOG_CHANNEL=stack

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=hiregallaxy
DB_USERNAME=root
DB_PASSWORD=

BROADCAST_DRIVER=log
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379


MAIL_DRIVER=smtp
MAIL_HOST=mail.hiregallaxy.com
MAIL_PORT=465
MAIL_USERNAME=support@hiregallaxy.com
```

This is the file overviews of this project and for further information go to the laravel.com's officials documentation about Laravel Project structure.

5. Python Setup :

- Go To Cpanel -> Click on Setup Python App
- Click On Create Application -> Create a folder in file manager
- Add the folder name in first section -> and the folder name after url
- Select the latest python version from top





Python version 2.7.16

Application root
It is a physical address to your application on a server that corresponds with its URI

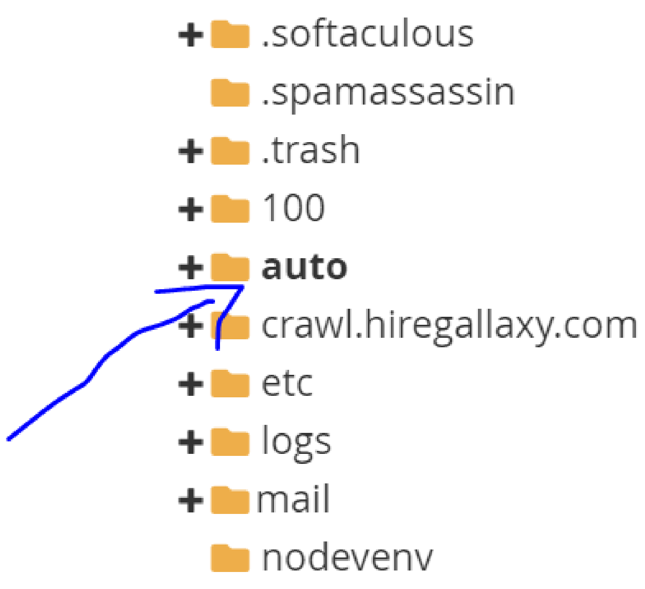
Application URL
It is an HTTP/HTTPS link to your application
hiregallaxy.com

Application startup file

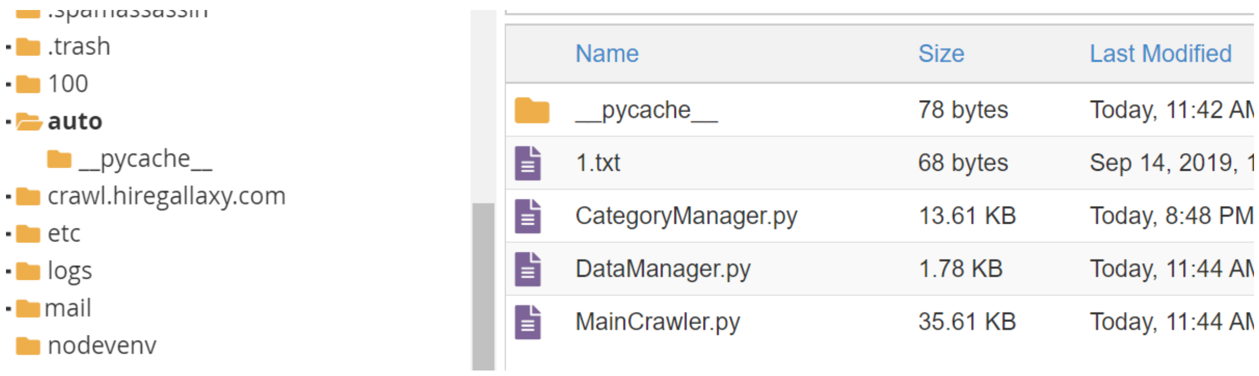
Application Entry point
Setup wsgi callable object for your application

6. Python File Overview :

- ★ Python crawler files are location in root folder -> auto
- ★ See the Screenshot below



+ .softaculous
.spamassassin
+ .trash
+ 100
+ **auto**
+ crawl.hiregallaxy.com
+ etc
+ logs
+ mail
nodevenv



.spamassassin
.trash
100
auto
__pycache__
crawl.hiregallaxy.com
etc
logs
mail
nodevenv

Name	Size	Last Modified
__pycache__	78 bytes	Today, 11:42 AM
1.txt	68 bytes	Sep 14, 2019, 1
CategoryManager.py	13.61 KB	Today, 8:48 PM
DataManager.py	1.78 KB	Today, 11:44 AM
MainCrawler.py	35.61 KB	Today, 11:44 AM

➔ In this version there are three files that are managing the the auto scraping
➔ DataManager.py
➔ CategoryManager.py
➔ MainCrawler.py

DataManager.py :

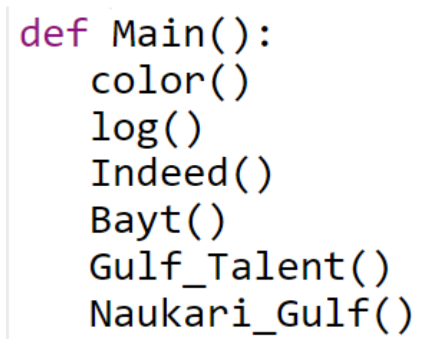
➢ This file manage the data efficiency that is scraped by crawler.It delete duplicate data.

CategoryManager.py

- This file handle the category of the job posted by crawler , if a jobs found to be different category then it change it to the right one.

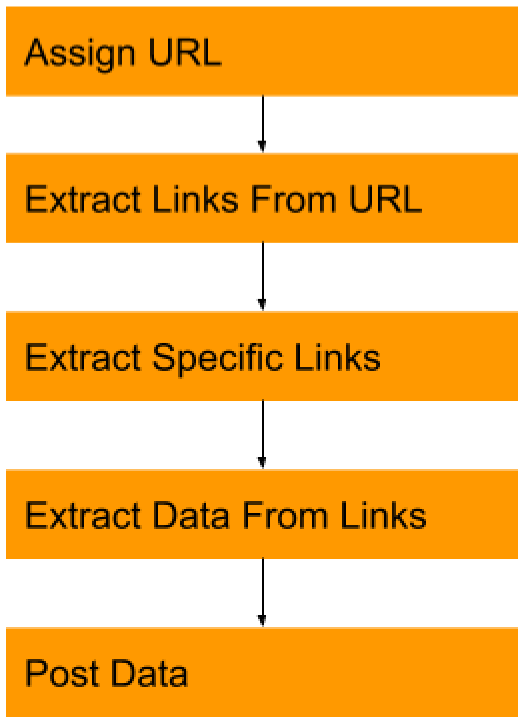
MainCrawler.py :

- This script handles the main crawling task.
- Below it overview of this script



```
def Main():
    color()
    log()
    Indeed()
    Bayt()
    Gulf_Talent()
    Naukari_Gulf()
```

Program Logic :


Assign URL
Extract Links From URL
Extract Specific Links
Extract Data From Links
Post Data