



By



Technical Guide

Product Version: 1.0.0

Table of Contents

<i>File Structure</i>	4
Controllers	4
File Structure	4
public/dist Folder	5
media-storage	5
<i>FAQs</i>	6
<i>Emails</i>	7
<i>Please Note</i>	8
<i>Thank You</i>	9

Server Requirements

Server requirements as given below:

- PHP >= 8.1.0
- OpenSSL PHP Extension
- PDO PHP Extension
- Mbstring PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension
- Ctype PHP Extension
- JSON PHP Extension
- BCMath PHP Extension
- Fileinfo PHP Extension
- GD Library (>=2.0)
- Curl
- MySQL 5.7 +

File Structure

While development we follow some of our rules to organize the code. For this we have made some changes to the Laravel framework files as follows:

Controllers

By default, Laravel placed all its controllers under **Http/Controllers** folder, our component controllers are placed **App/Yantrana/Components/** in their respective component

File Structure

All the component code is placed under the folder called **Yantrana** under the **app** folder. This folder contains other folders as follows:

- **__Laraware** – a folder contains a core file for the various classes which mostly extends Laravel classes & helper, which are restricted to change while development.
- **Base** – is also contains the base files for various classes which mostly extended using **__Laraware** classes.
- **Support** – contains various & common helpers & classes.
- **Services** – contains various & common required service classes.
- **Components** – This is main folder which contains all the business logic in to its own modules as follows:

- **Component Folder**

Each component has its own folder structure as follows:

- **Models** – Contains all the model files.
- **Controllers** – contains all the controller files.
- **Repositories** – contains all the repositories which treated as data-layer.
- **Requests** – contains the request files mostly for forms.
- **Blueprints** – contains the interface files for required classes.

- Engine.php – Its treated as most important file for the component

Each component can have multiple sub components.

public/dist Folder

This folder contains all the frontend related files like JS, CSS etc.

In the CSS & JS folder you will find files named like: **application.js**, **vendor-manage.css** etc.

For the application 2 types of files are there one is minified (.min) & other one is normal non-minified files (.src). By default, application uses **.min** files & called in to below named views which are placed under **resources/views** using **__yesset** helper functions, which dynamically finds the files on given file named & path contains *****.

media-storage

This folder stores all the uploaded media of the files like images etc. if not other storage options configured.

For the most other things you can consider **Laravel PHP Framework** documents at below link:

<https://laravel.com/docs/10.x>

Question:

How I can change homepage?

Answer:

You need to edit **outer-home.blade.php** file under resources/views folder.

Question:

I am using shared hosting and want to target public folder without public keyword in address bar?

Answer:

You need create .htaccess file with following code and place it into base folder (parent of public folder)

```
<IfModule mod_rewrite.c>
  RewriteEngine On
  RewriteRule ^(.*)$ public/$1 [L]
</IfModule>
```

Emails

For some action system uses email by default it uses php mail function as driver you can change it using `.env` file for more info see <https://laravel.com/docs/10.x/mail>

System uses `resources/views/emails/index.blade.php` as main view you can manage header & footer from here. Also, all the email view files are stored in `resource/views/email` folder.

Please Note

Here is some information you should aware of.

1. If you set `APP_DEBUG=true` in the `.env` file. the system also logs lot of information in to the **Browser's Console** panel for debugging, so please keep it `false` for the **production** uses.
2. When the `APP_DEBUG=false` in the `.env` file. the page's markup automatically minified
3. Admin Email is sent in **English** Only.
4. For the added protection sensitive information like username, passwords are sent encrypted to the server using OpenSSL, when form gets submitted When form has `data-secured=true` the form inputs get encrypted client side and gets decrypted server side.

Thank You

This information would help you to customize understand application better

Once again thank you so much for purchasing this product. I would be glad to help you if you have any questions relating to this product. No guarantees, but I will do my best to assist.

We are small team of Mobile and Web Designer & Developers, here are some of the Great tools and technologies we use day2day for the development: HTML5, CSS3, jQuery, PHP, Laravel, Flutter etc.

We also build **iOS & Android Application**.

If you need any Customization Work / Support on this Product or having any project for us. Please feel free to email us at contact@livelyworks.net

Thank You & Good Luck



livelyworks.net | contact@livelyworks.net | @livelyworks | <https://github.com/livelyworks>