**EASY PIE PHP FRAMEWORK 2.0.1**

Welcome to the easy pie php framework developed by Wongani Kaluwa to make a php life easier to use. Feel free to give feedback to this development via github, your suggestion will be a helping hand in refining this piece of art.

1. SETUP

The easy pie framework is a very easy to use and setup. To begin clone the easypie github repo or download the zipped project folder from any easypie source. Unzip the project in your htdoc, www folder on your xamp or wamp server document root directory. After that you are ready to begin the easy pie framework.

2. Folder and File Structure.

assets : contains all js, css, vendor files

routes: contains routes file for your routing system

src: contain the application controller and model logic

system: contains the framework source code: edit at your own risk

vendor: contains php frameworks installed by composer

views: contains your templates or html layouts

.settings : contains application local hosted settings or setup, eg. Database connections, application name, and other important variables

.settings-server : contains settings available to the production server, these are only usable on server, if a variable is not set on this one, it will be loaded from default .settings file instead

easypie : Command line Interface for the framework

index.php : application Entry Point.

3. ROUTING

Routing is how the framework handles the url in your browser. For example, each url is supposed to be defined before used other wise a load404 will be executed instead.

Example. Find and locate index.php stored in the routes folder.

The routes in this framework utilizes the Routes class, so always create an object and define get, post methods of it. Enough talking lets code.

Example 1:

Assume you would like to have a <http://localhost/easypie/home> this is however translated as a /home route so you need to define this in your routes before use.

$router->get('/home', 'views/home.php');

For more examples on usages, check the project default routes before use.

**COMMAND LINE INTERFACE**

This is a tool inbuilt with the easypie framework that allows automation when creating controllers, models and database migrations.

* Cleaning a project: Note this deletes any user generated code and leaves the project blank

Command: php easypie clean

* Creating models from database

Command : php easypie create-models model\_name\_1 model\_name\_2

The model\_name\_1 is a table from your database. This will automaticcaly generate ready to use models that will interact with your databases easuly.