Version 1.0

28-07-2018

PROJECT DOCUMENT

hashan kanchana kulasinghe

hashk99@gmail.com

# PROJECT DOCUMENT

*Project is explained in 4 separate sections.*

* Installation Guidance
* Folder and File Architecture
* Technologies and Plugins Used
* System Features

**INSTALLATION GUIDANCE**

1. Install the database script using **database\_script.sql** file.
2. Find the **constants.php** file and update the relevant variables
   1. **BASE\_URL** – replace with the server path
   2. **BASE\_FOLDER\_PATH** – directory path
   3. Database details in **DB\_HOST**, **DB\_NAME**, **DB\_USERNAME**, **DB\_PASSWORD**
   4. **GMAP\_KEY** – Google map API key to access relevant libraries
   5. **COMPANY\_EMAIL** – to get the contact form email
   6. **ENVIRONMENT** – PRODUCTION or LOCAL (to show/hide errors)
3. Always point the accessing url to **public** folder. (eg:- <http://localhost/myproject/public/>)

**FOLDER AND FILE ARCHITECTURE**

Project is developed using pure php functions and I have managed to build a simple yet clean framework to minimize the work stuff and changes.

* All the requests are route to **index.php** in public folder using .htaccess file
* **Bootstrap class** in the **index.php** file wrote to identify the request and call the necessary class and method. Just like the **MVC** structure but without models.
* I have used **gulp JS** library to maintain frontend files and libraries. Also **npm** as the frontend **package manager**. **Gulp run** command will concat all the css files in the root->frontend->css folder and will make a single file as styles.css in public->assets->css folder. Same goes for **Java Script** files.

**FOLDERS**

* Root->frontend
  + Keeps all the css, js and fonts in separate folders.
* Libs
  + Database.php
    - Create and return the mysql pdo database connection.
  + Functions.php
    - Common functions for the system (eg:- return relevant view using loadView method)
* **Node modules**
  + Front end libraries using npm
* **Pages**
  + Files inside this folder work as a controller. For each page, there is a separate file with the name you access. Second parameter works as the method.
    - Eg:- request -> localhost/project/contact/contactform\_post
      * This request will initiate the Contact class in the pages->contact.php file and will call the method contactform\_post with given POST data.
    - **404.php** – shows when there is a broken url request
    - **Blog.php** – blog page in the system
    - **Home.php** – index page in the system
    - **Projects.php** – projects page in the system
    - **Services.php** – services page in the system
    - **CommonPage.php** works as the parent class of each page class. All the **page controllers are extended** this class to have common behaviors such as the **database connection**
    - Pages folder **do not have** any frontend html view. All the html views are handled in a separate section
* **Public**
  + Folder where all the requests are handled. Also public folder has the necessary front end libraries and images
* **Views**
  + Views folder has all the project html files. I have managed to use the **same header and footer** files to avoid the **duplicate** for each page.
  + **Layout.php** file will require the **header.php , footer.php** and required page content so it won’t get duplicate for each

**TECHNOLOGIES AND PLUGINS**

* **FRONT END**
  + Node package manager
  + Gulp js
  + Bootstrap css framework
  + Animation.css
  + Fontawesome.css for icons
  + prettyPhoto.css for the image popup
  + jquery
  + isotope.js to handle a nice grid of elements in projects page
  + wow.js for advanced animations
* **BACKEND**
  + Php 5.6
  + PDO extension with mysql
  + .htaccess
  + Prepared statements for security enhancements

**SYSTEM FEATURES**

* Show news from database in the homepage
* Show projects from database and filter by categories
* Show blog posts from the database with user and comments count
* View all the comments by clicking read comments for each post
* Show recent comments in the right section
* Google map features
* Contact form to submit and have an email

Thank You !