

Doctor Chamber Management System

Developer Manual

Copyright | 2017
DBCINFOTECH

Support - send us a message from : <https://codecanyon.net/user/dbcinfotech>

Table of Contents

- Auto Installation
- Manual Installation
- Source Code Structure

Auto Installation

1. After you have downloaded the application from codecanyon, unzip it and put it on your server. You will find two folders 'documentation' and 'doorchamber' after you have unzipped the file. You should put the files and folders within the 'doorchamber' folder in your server. You need not to upload the documentation folder.
2. Let's say you have uploaded the files to a folder named 'chamber' in public_html folder of your server. Go to the browser and enter <https://yourservername.com/chamber>
3. You will get an auto installer loaded on your screen. Follow the steps mentioned in the auto installer and you will be good to go.

Manual Installation

Follow the steps carefully and chronologically to perform a successful install of the application. Any changes in the steps can bring issues and the installation might fail.

1. Download the application from codecanyon. Make sure you have downloaded the latest version
2. Upload the downloaded zip file to your server (public_html). You can upload the zip into a subfolder if you want.
3. Unzip the application in your server.
4. Go to your server's MySQL databases and create a database which this application is going to use. Keep in mind the database name you just created
5. Create an user for the MySQL database and add this user to the database you have created previously with all privileges
6. Open the file /application/config/database.php and provide the following informations

'hostname' => 'localhost',

'username' => 'the username of the user you just created',

'password' => 'the password you have given to the user',

'database' => 'the name of the database you created for the

application',

7. Go to your server phpmyadmin and import install.sql from your downloaded application (/uploads/install.sql) in your created database for this application.

8. Open the file /application/config/routes.php and change default controller from 'install' to 'login'

9. Now hit the link of this application and you will be good to go.

Default doctor email - doctor@example.com

Password - 1234

10. Go to settings and put your purchase code and save

Source Code Structure

The application is developed on Codeigniter PHP framework and it uses MySQL database to operate. The application uses php version of 5.6

The source code completely follows a MVC structure where the database interactions are handled through the model files, the view interactions and operations are handled through the controllers and the web pages are handled through views

Models

- Appointment_model.php - responsible for all the database interactions with 'appointment' table
- Billing_model.php - responsible for all the database interactions with 'invoice' table
- Chamber_model.php - responsible for all the database interactions with 'chamber' table
- Email_model.php - responsible for all the mail sending operations
- Patient_model.php - responsible for all the database interactions with 'patient' table
- Prescription_model.php - responsible for all the database interactions with 'prescription' table
- Settings_model.php - responsible for all the database interactions with 'settings' table
- User_model.php - responsible for all the database interactions with 'users' table
- Validation_model.php - responsible for all the form validation operations

Controllers

There is only one functional controller Doctor.php which controls the whole application through the help of models and helper functions

Views

View files are responsible for what you see on the webpage. These files contains all the html codes that renders an UI on page. This folder can be found at /application/views/.

- Index.php - this is the main view file of the application which makes the other view files dynamically load on this page
- Includes_top.php - this file contains links to all the stylesheets used in this application
- Includes_bottom.php - this file contains links to all the javascript files and plugins used in this application
- The folder 'doctor' contains the view files for the main body of the webpage which are dynamically loaded through index.php file

Assets

- Assets folder contains all the javascript and css files which have been used in this application to handle the user interface and many javascript events. It is suggested not to edit the third party plugin libraries without knowing what you are doing. Any changes done unknowingly can drastically alter the functionalities of the main application