CHAPTER III

PROBLEM ANALYSIS

III.1 MVC Pattern of CodeIgniter

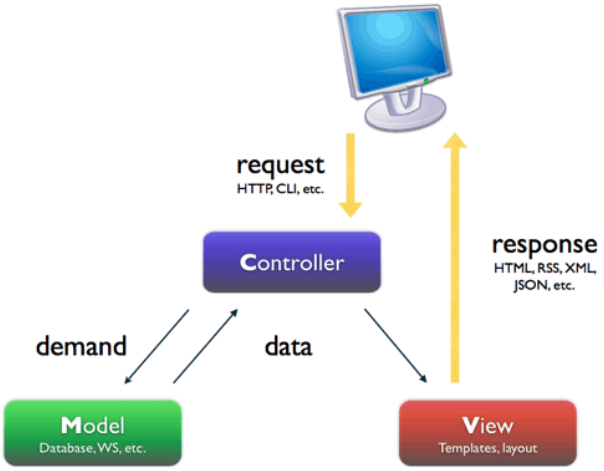

Figure 3.0 MVC Pattern

Source: http://stackoverflow.com/questions/5966905/which-mvc-diagram-is-correct-web-app

The MVC design pattern is good for web application because they split the database function, the function code and display separately and structured. With separated folder, the function is neater and doesn't look cluttered rather using manual function. Rather than include the file into the code like PHP Native usually do. The function separated with three folder, there are: [1]

1. Model

   Model is the part of the CodeIgniter system that manages all query related to data. Model usually used to code the query of CRUD (Create Read Update Delete) from database. The Model is responsible of all query database application.

2. View

   View is the part of CodeIgniter system that display the web. View used to display the template that created by Developer. It also display the code of query. As example we want to display product of today. Result of the code can be seen at View folder.

3. Controller

   Controller is the part of CodeIgniter system that manage all of logical function related to Model and View. Controller being a bridge between Model and View, because Controller has a function to manage the code which show the result of logical function. Controller also be a place to process the algorithm to show the result at the view folder.

III.2 CodeIgniter Controller Scope

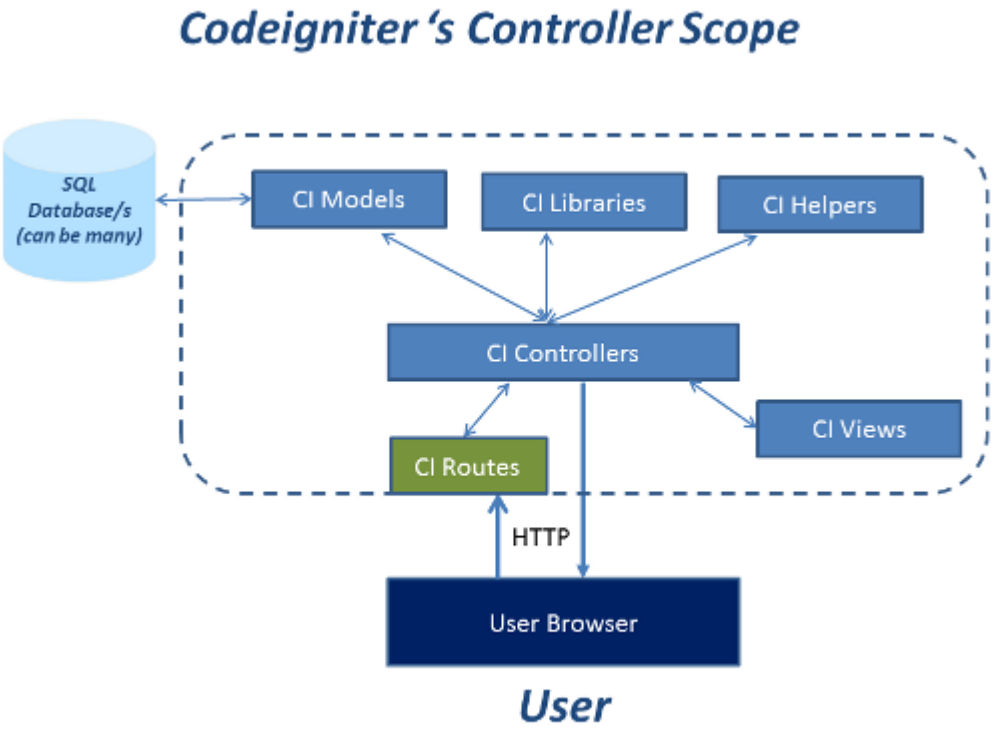

Figure 3.1 CodeIgniter's Controller Scope

Source : www.packtub.com

Code Igniter has a complex building. That's the reason why the security level in the CodeIgniter is better than using PHP Native. MVC Pattern connected with each other like the picture above. It show the Controller are the main processor of the program because Controller is the main function that display all of logical function. Without Controller, any libraries, model, and views can't be shown to the user. Several relationship will be describe below.

1. Model – SQL Database

   Model as the query function will call data appropriate the query code. As example, if the code are display the data. The database will bring the query request, that's data.

2. Controller – Libraries

   Controller as the logical function need libraries to make the program better. As example if we want use one template to some of template to make the program easy to code, we can using template library to make the template. The function of libraries is same like common libraries, to make developer easier when making the program with the help of library.

3. Controller – Helper

   Helper is a logical function that used by developer to make the code to use in some of views file. We can change the format date from 12//12/2016 into Tuesday, December 12 2016 with helper file. The objective is to be a template of code and developer can call it when needed. The perspective are same like Object Oriented Programming.

4. Controller – Routes – User Browser

   Routes has a function as a bridge between the logical and user browser. Its mean the logical function that displayed to the user will going through a bridge called Routes. Routes also being a link of HTTP in user browser.

   More complex about the architecture is when user want to see the web, they will input the address, after user input it will call the routes to check are the link is available at the controller. Controller will check the libraries and helper which needed in the view. Controller will call the libraries and helper, and also the model to get the data. Model will check data from query code that called by controller and return the data to the controller, after that the data will be displayed to the user via routes.