MVC Framework

Introduction:
MVC stands for “Model View Controller. The MVC is an architectural pattern that separates an application into three logical components: The Model, The View and The Controller.

Model:
Fetch data from the database and send to the controller. Interact with the database.

Controller:
Handles the business logic. Monitor users action and send to model.

View:
Display data and captures User actions. Send user action to controller.

Use:
Without the use of MVC you code can be more complicated and dense. After the certain period of time you cannot find errors and issues easily. The durability of your code may be less. MVC allows you to separate business logic from your presentation layer. This “Separation Of Concern” allows you to quickly find and edit portions of your code. It also enables easy reuse of your UI components across your system.

Advantages Of MVC Pattern:
- ✔ Reduce code flexibility.
- ✔ Re-usability.
- ✔ It provides a clean separation of concerns.
- ✔ Manageability.
- ✔ Easy to find issues and errors.
- ✔ Readability.

Disadvantages Of MVC Pattern:
- ✔ The main disadvantage of MVC architecture is that it is not suitable for smaller applications which has adverse effects in the applications performance and design.
- ✔ The complexity is high to develop the application using MVC pattern.

Framework:
- ✔ A framework is an essential supporting structure of building something.
- ✔ A framework provides you standard solutions to typical problems.
- ✔ Technically the framework is the collection of classes and Methods which helps you develop web applications.

Framework Advantages:
- ✔ Provide an uniform and proper structure to your web application.
- ✔ Speeding up the development process.
- ✔ Reusing the code in similar projects will save the developer a substantial amount of time and effort.
- ✔ The code (usually) works, it is already tested.

Framework Disadvantages:
- ✔ If you want to extend the functionality you have to understand other people codes (OPC).
- ✔ If the framework contains the security hole and an exploit is available you site is immediately vulnerable, but you may not have the knowledge to fix it yourself.
- ✔ Meaningless to apply to small web applications.

MVC Framework

Code-Igniter:

- ✔ Code-Igniter is a powerful PHP framework, with a very small foot-print build for developers who want simple and elegant toolkit to create full features web applications.
- ✔ Code-Igniter is the second most popular web framework among PHP developers.
- ✔ It is a lightweight powerful PHP framework that provide simple and elegant platform to create full featured web applications.

Use:

- ✔ Code-Igniter is loosely based on the popular model view controller (MVC) development pattern.
- ✔ Easy to understand and extend.
- ✔ Large and active community.
- ✔ Good for start-up.
- ✔ Light weight.

Code-Igniter Advantage:

- ✔ Easy and hassle-free migration from server hosting to server hosting.
- ✔ Easy handling and customizing.
- ✔ A new functionality can be applied without effecting the customizing at all.
- ✔ Offers easy management and flexibility with MVC based framework.
- ✔ Provides easier configuration and customization of configuration files.

Code-Igniter Disadvantages:

- ✔ It is PHP based and not very object-oriented in some parts.
- ✔ Company driven instead of community driven.
- ✔ Not provide pre-defined database migration.

Code-Igniter Features:

- ✔ Free from complex structures and development.
- ✔ Depiction of straightforward solutions.
- ✔ Provision of broad compatibility along with standard hosting accounts.
- ✔ Configuration and normal coding rules are not required.

Prerequisite:

- ✔ Knowledge of core PHP.
- ✔ Basic knowledge of OOP comcepts.
  - What is Class?
  - What is functions/methods?
  - What is constructors.
- ✔ Knowledge of MVC pattern.

Code-Igniter Structure:

- ✔ The application folder.
- ✔ The System Folder.
- ✔ User Guide.

MVC Framework

The "application" Folder:
The application folder contains the application you are currently building. Basically this folder contains models, views, controllers, helpers classes and classes extensions. This is the folder where you will work for your web application project.

The "system" Folder:
The system folder is where all the action happens. This folder contains all the Code-Igniter code of consequence, organized into various folders.

The "user guide" Folder:
Documentation of Code-Igniter use.

Controller:
A controller is a simple class file. As the name suggest it controls the whole application by URI. To create a controller go to (application/controller) folder and create a class file. Code-Igniter provides a default controller file which helps us define our own controller, that default file is called (CI_Controller). Always keep first letter capital while defining controller. File name and Class name should be same.

Model:
To create a model go to (application/model) and create a model file. According to CI naming convention, name like (Xyz_model) is compulsory. Keep first letter capital. CI provides a default class to define our custom models by extending (CI_Model) class. Model communicates with the database in database driven application written in CI.

View:
To create a view go to (application/views) folder and create a file. Load view in controller.

Libraries:
The essential part of the Code-Igniter framework is its libraries. It provides a rich set of libraries which indirectly increase the speed of developing an application. The system libraries is located at (system/libraries) folder. All we need to do is to load the library that we want to use ($this→load→library('library_name')) is the syntax of loading a library in a controller. There are some inbuilt CI libraries such as (sessions, email, pagination, file upload) etc. To create the custom library you will access (application/library) folder. While loading the library in controller class, you should specify the library class name in small or capital letters because Code-Igniter doesn't care about that. When calling the method of a library you should use lowercase library class object name and then method name.

Helpers:
Helpers as the name suggest helps you with task. Each helper class is simply a collection of function sin a particular category. There are many helpers such as (URL, Text, Cookie) available in Code-Igniter. To create custom helper we need to access (application/helper) folder and create a custom helper file there. To load the helper class web need to write like this ($this→load-helper('helper_name')).

MVC Framework

Routing:
Routing in Code-Igniter manages the URL and enable developers to rewrite their custom URL. It identifies the URL and perform the certain actions.

Reserved Names:
Code-Igniter uses some predefined keywords which can't be used by the developers. Such as:

- Controllers. (Controller name)
- load_class(). (function names)
- $config. (Variable name)

Project: Build an application called Calculator.

CI Classes:
Code-Igniter provides some pre-made classes to perform a certain actions. It provides scalability, flexibility to the developer. In terms of CI, these classes are exhibited as libraries. There are so many CI classes available. Some of them are:

- Calendaring Class.
- Config Class.
- Email Class.
- Encrypt Class.
- File uploading Class.
- Form validation Class.

E-Commerce:
In its simplest form e-commerce is the buying and selling of products and services by business or consumers over the world wide web. People use the term "e-commerce" or "online shopping" to describe the process of searching for and selecting products in online catalogues and then checking out using the credit card and encrypted payments. Some e-commerce sites are (amazon, ebay and darazpk).

Cart:
In online marketing, shopping cart is a piece of e-commerce software on a web server that allow visitors to select items for eventual purchase. Analogous to American English terms "shopping cart". Simply called a Basket. Code-Igniter provides built-in shopping cart library. The cart class permits items to be added to a sessions that would be active while a user is browsing the site. These items can be retrieved and displayed in a standard shopping cart format, allowing the user to update the quantity or remove items from the cart. Shopping cart stores the products which the user willing to buy. Product Name, Product Price and Product Quantity.