**PHP MVC框架演化的实证研究**

**摘要：**Web技术的商业，社会和教育重要性极大地增加了Web编程/脚本编写的研究活动。 已经提出了几种用于编写PHP代码的方法，例如面向对象编程（OOP），程序PHP编码和模型视图控制器（MVC）模式。 模型视图控制器（MVC）是开发PHP应用程序最强大的方法之一，有许多框架，如Laravel，Symfony，CodeIgniter，CakePHP等。但是，在框架中选择最佳MVC框架是程序员关注的问题以及项目经理，尤其是在管理大型应用程序时 因此，需要绩效评估标准。 本文讨论了基于MVC的最著名的PHP框架，评估了它们的性能，发现Laravel优于其他MVC框架，因此Laravel被认为是最适合未来Web技术的PHP框架。

**关键词：**MVC, laravel, PHP Framework, cakePHP, CodeIgniter, Symfony

1. **简介：**

基于Web的应用程序的互联网的快速发展表明对编码方法的可靠性，可伸缩性，安全性和可维护性的更高要求。 PHP，一种用于Web的脚本工具，可实现动态交互式Web开发，如直观，快速编译，跨平台，开源，灵活性以及所需的最小设置。 这成为重要的Web开发语言之一，因此，PHP是Web世界中最强大的编程语言之一。 一些开发人员选择部署基于PHP的应用程序，将所有问题（如数据访问，业务逻辑和数据表示层）放在一起。这反过来又会产生开发问题，特别是对于大型项目。 为了解决这个问题，MVC设计模式带来了一种基于每个层活动将层中的代码彼此分离的有效方法。

MVC设计模式是开发应用程序（如CakePHP，Codeigniter，Laravel，Symfony）的有效方法。 MVC的主要方法是将应用程序分散到单独的层中，该层可以单独工作并产生相同的结果。 使用MVC模式的优点是 –

标准，一致性和可预测性

软件组件或构建块，以便开发人员可以共享和重用代码

一种模型或标准体系结构，可以轻松查看整个系统的工作方式

库，类和函数中可重用且经过全面测试的代码

使用架构模式的结构良好的代码

安全性，互操作性和维护

虽然基于MVC的框架（cakePHP，laravel，codigniter）具有许多优点，但是，选择最佳PHP框架仍然是一个问题。 这是因为所有框架都没有涵盖Web应用程序的所有方面。 本研究评估了基于MVC设计模型和性能的大多数着名的PHP框架，并为未来的Web开发提出了最有效的PHP MVC框架。

1. **构造和方法：**

模型视图控制器（MVC）方法的操作是将代码的不同部分分散或分离成层，例如视图，数据访问，控制用户的请求以及向相关层的转发请求。

MVC模式的标题是三个核心部分的整理：模型，视图和控制器。 完整和正确的直观表示它是数据的单流布局，它是如何在每个组件之间传递的，最后是每个组件之间的关系如何工作。

1. **模型**

模型是给予整体设计中使用的数据的永久存储的名称。 它必须允许访问数据，或者收集和写入数据，并且是整个模式中View组件和Controller组件之间的桥梁。 该模型的一个重要方面是它在技术上是“盲目的” - 由此，模型与传递给View或Controller组件时数据发生的变化无关。 它既没有打电话也没有寻求该组成部分其他部分的回应; 其主要目的是将数据处理到其永久存储中，寻找并准备要传递到其他部分的数据。

模型不能简单地假设为数据库工具包，或者是处理数据处理的另一个系统的网关。 模型代表数据本身的守门员，不问任何问题，但接受所有请求。 通常是MVC系统中最复杂的部分，Model组件也是整个系统的巅峰之作，因为没有它，Controller和View之间就没有连接。

1. **视图**

视图是一个模块，用于查看从模型中请求的数据并确定其最终输出。 传统上，在Web应用程序中使用MVC进行开发，View是生成和显示HTML的系统的一部分。 View还会点燃用户的反应，然后用户继续与Controller进行交互。 基本示例是由View生成的按钮，用户单击该按钮并在Controller中触发操作。

**关于View的误解**

关于View组件存在一些误解，特别是Web开发人员使用MVC模式来构建他们的应用程序。 例如，许多人认为View与模型没有任何关联，并且View显示的所有数据都是从Controller传递的。 实际上，这种流程完全忽视了MVC模式背后的理论。 Fabio Cevasco的文章The CakePHP Framework：Your First Bite 在CakePHP框架中演示了这种混淆MVC的方法**。**

为了正确应用MVC架构，模型和视图之间必须没有交互：所有逻辑都由控制器处理。此外，视图作为模板文件的描述是不准确的。 View实际上不仅仅是一个模板，现代MVC启发的框架已经把这个观点混淆到几乎没有人真正关心框架是否真正遵循正确的MVC模式。同样重要的是要提到 视图部件永远不会由Controller提供数据。 视图和控制器之间没有直接关系，它们之间没有模型。

1. **控制器**

第三个组成部分是控制器。它的工作是处理用户提交的数据以及相应地更新模型。控制器可以概括为信息收集器，然后将其传递给要组织存储的模型，并且除了收集用户输入之外不包含任何逻辑。 Controller也只连接到单个View和单个Model，使其成为单向数据流系统，在每个数据交换点都有握手和签收。控制器仅在用户首先与View交互时执行任务，并且每个Controller功能都是触发器，由其引发

用户与View的交互。开发人员最常犯的错误是将Controller混淆为网关，并最终为其分配View应该执行的功能和职责（这通常是同一个开发人员将View组件混淆为模板的结果）。此外，分配Controller函数是一个常见的错误，这些函数使其自行负责处理，传递和处理从Model到View的数据。尽管如此，MVC模式关系应保持在Model和View之间。

**MVC设计框架**

1. **Codeigniter:**

CodeIgniter是一个开源的快速开发Web应用程序框架，用于使用PHP构建动态网站。 它的目标是通过为常用任务提供丰富的库集以及访问这些库的简单接口和逻辑结构，比从头编写代码更快地开发项目。 CodeIgniter的第一个公开版本于2006年2月28日发布，最新的稳定版本2.1.4于2013年7月8日发布，CodeIgniter基于流行的模型 - 视图 - 控制器开发模式。 虽然视图和控制器类是CodeIgniter下开发的必要部分，但模型是可选的。 与其他PHP框架相比，CodeIgniter通常以其速度而闻名**。**

1. **CakePHP:**

CakePHP是一个开源的Web应用程序框架。 它遵循模型 - 视图 - 控制器（MVC）方法，并以PHP编写，以Ruby on Rails的概念为模型，并在MIT许可下分发。CakePHP使用着名的软件工程概念和软件设计模式，如配置约定，模型 - 视图 - 控制器，活动记录，关联数据映射和前端控制器。

1. **Symfony:**

Symfony是一个用于MVC应用程序的PHP Web应用程序框架。 Symfony是免费软件，根据MIT许可证发布。 symfony-project.com网站于2005年10月18日推出**。**

1. **Laravel:**

Laravel是一个免费的开源PHP Web应用程序框架，专为MVC Web应用程序的开发而设计。 Laravel是在MIT许可下发布的，其源代码托管在GitHub上。

1. **结论：**

本文对PHP框架的主要MVC模式进行了实证研究。从评估四个框架获得的结果：CodeIgniter（CI）Symfony CakePHP和Laravelusing标准，如每秒请求数，系统负载平均值，内存使用量，函数调用数和所需文件数以及每个框架中的可用设施显示theLaravel优于其他MVC框架。对于像CakePHP这样的其他人来说，每秒高达3000的请求被记录为低至每秒750个请求。所有其他参数获得的结果包括存储，函数调用，文件数量，响应时间e.t.c.表明该层具有巨大的Web应用程序开发灵活性，它为程序员提供了更多的功能，使得所有Web程序员都可以接受直观，快速编译，跨平台，开源，灵活等不同标准。它可以轻松迁移，丰富的库，模板系统，雄辩的ORM和广泛的社区支持，有助于顺利开发应用程序。 laravel的所有标准和设施证明，laravel是部署下一代基于PHP的Web应用程序的最佳选择。

**AN EMPIRICAL STUDY OF THE EVOLUTION OF PHP MVC FRAMEWORK**

**Abstract:** Commercial, social and educational importance of web technology has tremendously increased research activities in web programming/scripting. Several methods for writing PHP codes such as Object Oriented Programming (OOP), Procedural PHP coding and Model View Controller (MVC) pattern have been proposed. Model View Controller (MVC) which is one of the most powerful method for developing PHP application has many variant such laravel, Symfony, CodeIgniter, CakePHP etc. However, selection of best MVC framework among the variants is of concern to the programmers as well as project managers, especially when managing big applications. Hence,performance evaluation criterions are required. This paper discusses the MVC based most famous PHP frameworks, evaluate their performance and it was found that Laravel outperforms other MVC framework, hence laravel is proposed as the most suitable PHP framework for future web technology.

**Keywords:** MVC, laravel, PHP Framework, cakePHP, CodeIgniter, Symfony

**1. Introduction:**

The rapid development of internet for web based application indicates a higher demand of reliability, scalability, security and maintainability of coding methodology. PHP, a scripting tool for web that enable dynamic interactive web development suchintuitive, compiled fast, cross platform, open source, flexibility as well as required minimal setup. This became one of the important web development language thus, PHP is one of the most powerful programming language in the web world. Several developers choose to deploy application based on PHPputting all the issues such as data access, business logic, and data representation layer together.This in turn createdevelopment problems especially for big projects. To solve this problem, MVC design pattern brings an effective ways to separate code in layers from each other based on each layer activities.

MVC design pattern is a proven effective way to developapplication such as CakePHP, Codeigniter, Laravel, Symfony. The mainmethods of MVC to spilt an application into separate layer that can work separately and produce same result. The advantage of using MVC pattern are-

• Standard, consistency and predictability

• Software components or building-blocks so that developers can share and reuse code

• A model or standard architecture that allows easy visualization of how the entire system works

• Reusable and thoroughly tested code in the libraries, classes and functions.

• Well-structured code using architectural pattern.

• Security, interoperability and Maintenance.

Although MVC based framework (cakePHP, laravel, codigniter) has number of advantages,however, selecting of best PHP framework is still a concern. This is because all of the framework does not cover all aspect of web applications. This study evaluate most famous PHP frameworks based on MVC design model and it performance as well as proposed the best efficient PHP MVC framework for future web development.

**2. Materials and Methods:**

The operation of Model View Controller (MVC) method is to spilt or separate the different parts of code into layers such as view, data access, controllinguser’s requests and forward request to relevant layers The MVC pattern’s title is a collation of three core parts: Model, View, and Controller. It is a single flow layout of data, how it’s passed between each component, and finally how the relationship between each component works

**a. Model**

The Model is the name given to the permanent storage of the data used in the overall design. It must allow access for the data to be viewed, or collected and written to, and is the bridge between the View component and the Controller component in the overall pattern. One important aspect of the Model is that it’s technically “blind” – by this, the model has no connection or knowledge of what happens to the data when it is passed to the View or Controller components. It neither calls nor seeks a response from the other parts of the component; its main purpose is to process data into its permanent storage, seek and prepare data to be passed along to the other parts.

The Model cannot simply be assumed as a database toolkit only, or a gateway to another system which handles the data process. The Model represents a gatekeeper to the data itself, asking no questions but accept all requests which comes its way. Often this most complex part of the MVC system, the Model component is also the pinnacle of the whole system since without it there will be no connection between the Controller and the View.

**b. View**

The View is a module where data, requested from the Model is viewed and its final output is determined. Traditionally in web application use MVC for development, the View is the part of the system where the HTML is generated and displayed. The View also ignites reactions from the user, who then goes on to interact with the Controller. The basic example of this is a button generated by theView, which a user clicks and triggers an action in the Controller.

**Misconceptions about View**

There are some misconceptions held about View components, particularly by web developers using the MVC pattern to build their application. For example, many mistake the View as having no connection whatsoever to the Model and that all of the data displayed by the View is passed from the Controller. In reality, this flow disregards the theory behind the MVC pattern completely. Fabio Cevasco’s article The CakePHP Framework: Your First Bite demonstrates this confused approach to MVC in the CakePHP framework.

In order to correctly apply the MVC architecture, there must be no interaction between models and views: all the logic is handled by controllers.Furthermore, the description of Views as a template file is inaccurate. The View is really much more than just a template, themodern MVC inspired frameworks have bastardised the view almost to the point that no one really cares whether or not a framework actually adheres to the correct MVC pattern or not.It’s also important to mention that the View part is never given data by the Controller. There is no direct relationship between the View and the Controller without the Model in between them.

**c. Controller**

The third component of the triad is the Controller. Its job is to handle data the user submits as well as update the Model accordingly. The Controller can be summed up as a collector of information, which then passes it on to the Model to be organized for storage, and does not contain any logic other than collecting user input. The Controller is also only connected to a single View and to a single Model, making it a one way data flow system, with handshakes and signoffs at each point of data exchange. Controller is only given tasks to perform when the user interacts with the View first, and that each Controller function is a trigger, set off by

the user’s interaction with the View. The most common mistake made by developers is confusing the Controller as a gateway, and ultimately assigning it functions and responsibilities that the View should do (this is normally a result of the same developer confusing the View component as a template). Additionally, it’s a common mistake to assign the Controller functions that gives it the sole responsibility of crunching, passing, and processing data from the Model to the View. Nonetheless, the MVC pattern relationship should be kept between the Model and the View.

**MVC Design Frameworks**

1. **Codeigniter:**

CodeIgniter is an open source rapid development web application framework, for use in building dynamic websites with PHP. Its goal is to enable to develop projects much faster than writing code from scratch, by providing a rich set of libraries for commonly needed tasks, as well as a simple interface and logical structure to access these libraries. The first public version of CodeIgniter was released on February 28, 2006, and the latest stable version 2.1.4 was released July 8, 2013 ,CodeIgniter is loosely based on the popular Model-View-Controller development pattern. While view and controller classes are a necessary part of development under CodeIgniter, models are optional. CodeIgniter is most often noted for its speed when compared to other PHP frameworks.

1. **CakePHP:**

CakePHP is an open source web application framework. It follows the Model-View-Controller (MVC) approach and is written in PHP, modeled after the concepts of Ruby on Rails, and distributed under the MIT License.[19] CakePHP uses well-known software engineering concepts and software design patterns, as Convention over configuration, Model-View-Controller,Active Record, Association Data Mapping, and Front Controller.

1. **Symfony:**

Symfony is a PHP web application framework for MVC applications. Symfony is free software and released under the MIT license. The symfony-project.com website was launched on October 18, 2005.

1. **Laravel:**

Laravel is a free, open source PHP web application framework, designed for the development of MVC web applications. Laravel is released under the MIT license, with its source code hosted on GitHub.

**3. Conclusion:**

An empirical study on major MVC pattern for PHP framework has been evaluated in this paper. The results obtained from evaluating the four frameworks: CodeIgniter (CI)Symfony CakePHP and Laravelusing criteria such as request per second, system load average, memory usage, number of function calls and number of files required as well as available facilities in each framework shows that theLaravel outperforms other MVC framework. A request per second of as high as 3000 was recorded forLaravelcompare to others like CakePHP with as low as 750 request per second. The results obtained for all other parameters such has storage, function calls, number of files, response time e.t.c. indicates thatlaravelhas huge flexibility of development of web application, it has some more facilities for programmers that makes it acceptable to all web programmer in terms of different criteria such as intuitive, compiled fast, cross platform, open source, flexibility. It enable easily migration, enriched library, template system, eloquent ORM and wide range of community support that helps to develop application smoothly. All the criteria and facilities of laravel prove that, it is of opinion that laravel would be the best choice to deploy next generation PHP based web application