

A COMPARATIVE STUDY ON WEB FRAMEWORK TECHNOLOGY

¹NIKITA DAROKAR

Shri Sant Gajanan Maharaj College of Engineering, Amaravati University, Shegaon
nikitadarokar18@gmail.com

²HARSHA GAWANDE

Shri Sant Gajanan Maharaj College of Engineering, Amaravati University, Shegaon
1harshagawande@gmail.com

³SAKSHI BAJAJ

Shri Sant Gajanan Maharaj College of Engineering, Amaravati University, Shegaon
sakshibajaj1444@gmail.com

³VAIDEHI GAWANDE

Shri Sant Gajanan Maharaj College of Engineering, Amaravati University, Shegaon
vaidehigawande0260@gmail.com

ABSTRACT: *Nowadays World Wide Web has turned into a whole new level of marketplace. The websites give a business strong online presence. The WWW is now a dynamic medium exposing a plethora of on-line services for every user. The web technologies are rapidly changing and developed new. This makes the decision of choosing one technology hard for the developer. That is why developer needs to keep up with latest trends. The comparison of dynamic Web programming framework technologies is the main goal of this paper. There are many web development frameworks available; choosing the one that suit our need is the hard part. In particular, this paper examines difference between asp.net, Django, code igniter, cake php, angular js. The comparison is based on the many factors like System architecture, performance, speed, API support, Cost estimation, etc.*

Keywords: frameworks, System architecture, Performance, API Support.

1. INTRODUCTION

A web application is a software application which runs on one or more than one computer or a machine. A web application communicates through a network or a server. In most cases, web browsers as a client are used to access web applications over a network, such as Internet. This provides the ability to update and maintain a program without deploying and installing software on client machines. Web applications are utilized in intranets, in many companies and schools, for instance. They are used for web mail, online retail sales, discussion boards, weblogs, online banking among others. Web applications are different from other applications because they do not need to be installed and hence have an advantage that it can be accessed and used by millions of people at the same time [1].

Because most computer operating systems have web browsers web applications are popular. Programmers can easily change a web application. Users don't go to install new software to ascertain these changes. As web applications run on Internet or Intranet, they have become an essential and important part of business recently. More sophisticated Internet and web application have been emerged as the number of users of Internet and World Wide Web is increasing day by day.

Dynamic elements such as event handling, processing of forms, performing of calculations have been included in order to extend the functionality of Internet browsers. VBScript or mostly used JavaScript are the script languages introduced to enhance the mostly static content of web pages with dynamic elements.

In order to provide interactivity to web applications, Dynamic HTML (DHTML), extensible HTML (EHTML) and XML are some of the technologies that are part of today's browsers. Web servers have become more flexible while responding to client that presenting the same content to all users. Server side scripting use in order to process the incoming requests by the server.

With the help of server side scripting considering the client's request web pages are prepared by application on the server. In HTML format, the information is submitted to the client which is the Internet's basic standardized language of communication. Several server side technologies exists today, example are ASP or ASP.Net, Java Servlets and JSP, PHP, Perl, Python. A web framework may is a software framework that's designed to support the event of web applications including web services, web resources and web APIs [7]. They provide a standard way to build and deploy web application on World Wide Web. They aim to automate the overhead associated with common activities performed in web development.

Comparing various web development tools and technologies is an important idea consideration for many web application developers. This in turn helps to decide which technologies and tools to adopt in developing new frameworks to simplify the web application development process[2] [3]

The comparison of dynamic Web programming framework technologies is the main goal of this paper. There are many web development frameworks available; choosing the one that suit our need is the hard part. In particular, this paper examines difference between asp.net, Django, code igniter, cake php, angular js. The comparison is based on the many

factors like System architecture, performance, speed, API support, Cost estimation etc.

2. LITERATURE SURVEY

Comparison is based on system architecture and implementation, orm techniques, performance. System architecture (Model View Controller (MVC) Model View Controller (MVC) is the type of design pattern. Most of the PHP frameworks and content management systems are based on MVC design pattern. MVC separates the application into three different layers, Model, View and Controller which makes the application very light. New features can easily be added in MVC based application. The layout of application can be easily changed.

2.1 Cake PHP

2.1.1. System Architecture

Cake PHP is a MVC based framework and was released in 2005, it is written in PHP but inspired by Ruby on Rails. Ruby is a cross-platform interpreted language. Rails is a web development framework which runs on the Ruby programming language. Ruby on rails also uses MVC design pattern. Cake PHP uses documented software engineering concepts and software design patterns as convention over configuration, Model View Controller, active records, association data mapping and front controller

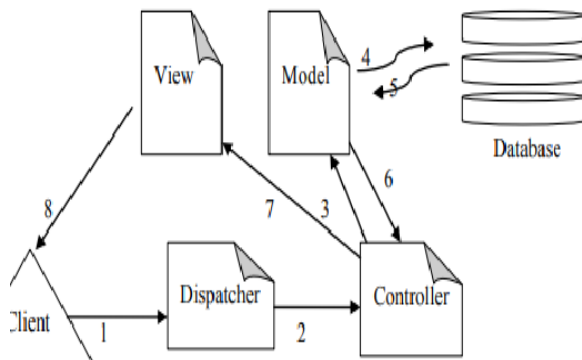


Figure 1: System Architecture of Cake PHP

2.1.2. Orm technique

Cake PHP follows ORM technique wherein object relational mapping programming that turns the info of various type systems in the database.

2.1.3. Performance

It provides much faster performance in case of small data sets and small projects but for larger projects it is not that much preferred as it does not perform data binding process

2.1.4. API Support

APIs are handled well in Cake PHP due to its code structure and most things implemented the right way. However, implementing APIs and frankly the Cake routing itself has been made unnecessarily difficult and people tend to stay away from Cake PHP due to this reason. [4]

2.2.1. System Architecture

CodeIgniter is also MVC based PHP framework, was written by Rick Ellis. CodeIgniter framework has some distinct features i.e. no restrictive coding rules, no need to learn template language, small but comprehensive libraries and thorough documentation. These features are suitable for small and medium sized application. In CodeIgniter, there is no database abstraction layer like object-relational mapping (ORM) in CAKEPHP. Due to absence of ORM in CodeIgniter framework, the database communication becomes complex and insecure.

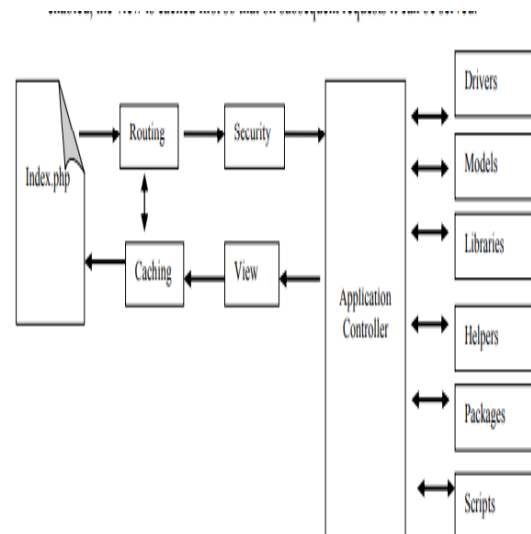


Figure 2: System architecture of CodeIgniter

2.2.2 Orm technique

CodeIgniter doesn't accompany ORM, so for the conversion, developers need to believe the third parties

2.2.3 Performance

It offers enhanced stability and support, it offers simple routing methods it allows you to cached website for improved performance and loading times.

2.2.4. API support

There was no built-in API support in 2.0. CI provided new base controller called Rest Controller. There is still no concept of middleware in CodeIgniter hence the application will not be based on latest programming practices on other accomplished frameworks. [5]

2.3. AngularJS

AngularJS may be a popular JavaScript framework for creating front-end single page web applications. It's designed to support dynamic views which makes browsing the page smooth like that during a native application. Important features are:

2.2 Codeigniter

2.3.1. System Architecture

Web developers were introduced to the Model-View-View Model (MVVM) architectural pattern. Understandably, this required a touch of a paradigm shift for several coders. Moreover, there was much confusion about whether Angular was really Model-View Controller (MVC) or Model-View-View Model (MVVM) Angular follows a component-oriented architecture. Having said that, Angular does share number of the concepts of both MVC and MVVM.

2.3.2. Orm Technique:

As most existing web development environments specialise in implementing ORM on the server side, this will end in performance and communication bottlenecks – to not mention increased infrastructure costs. By offloading some of these ORM tasks to AngularJS, you'll parallelize many of those tasks and reduce overall server load, in some cases obviating the necessity for the server entirely.

2.3.3. Performance

Performance is one among the most important problems with the original angular JS. It's due the factor that made Angular JS so popular. To realize two way data binding it relies on digest cycle for keeping views in sync with their underlying data models. Angular Js functions by augmenting event handler with process called dirty checking and every of scoped variable are compared their previous values.

2.3.4. API support

Angular JS may be a toolset for building the framework most suited to your application development. It's fully extensible and works well with other libraries. [6]

2.4. Django

Django is a high-level Python based free and open source Web framework which helps in rapid development of an application and it's open source framework.

Tools Integrate with Django are: Python, Pycharm, Laravel, sentry

Django tools

Django-debug-toolbar may be a very handy tool that gives information about what your code is doing and time consumption for it. It show all the SQL queries that page is generating, and how much time required to execute it. Third-party panels also are available for the toolbar, which will (for example) report on cache performance and template rendering times.

2.4.1. System Architecture

Django follows Model-Template-View (MTV) architectural pattern. It is maintained by Django Software Foundation(DSF).

In a traditional website, a internet application waits for HTTP requests from the web browser. When an invitation is received the application works thereon consistent with the request URL and possibly information in POST data or GET data. Depending on what is required it read or writes information from a database or performs other tasks required to satisfy the request. This will then return a response to the web browser, often dynamically creating an HTML page for the browser to display by inserting the retrieved data into placeholders in an HTML page. Django web applications typically group the code that handles each of these steps into separate files:

- 1) **URLs:** Process requests from every single URL, it's more maintainable to write down a separate view function to handle each resource. A URL mapper is employed to redirect HTTP requests to the acceptable view based on the request URL. The URL mapper also can match particular patterns of strings or digits that appear during a URL and pass these to a view function as data.
- 2) **View:** A view is a request handler function, which receives HTTP requests and returns HTTP responses.
- 3) **Models:** It's python object which define the structure of an application's data, and supply mechanisms to manage insert, modify, delete and query records in the database.
- 4) **Templates:** It's a document defining the structure or layout of a file , with placeholders used to represent actual content. A *view* can dynamically create an HTML page using an HTML template, A template can be used to define the structure of any sort of file.

2.4.2 ORM Technique

ORM is extremely efficient and it's tightly coupled with the Django framework. Django ORM is suitable for handling low-medium complexity queries. Although, some consider **SQL Alchemy ORM** to be a better option. It also provide **migrations feature to Django** . The ORM has many benefits and Querysets. Django is that the only framework which is complete in itself.

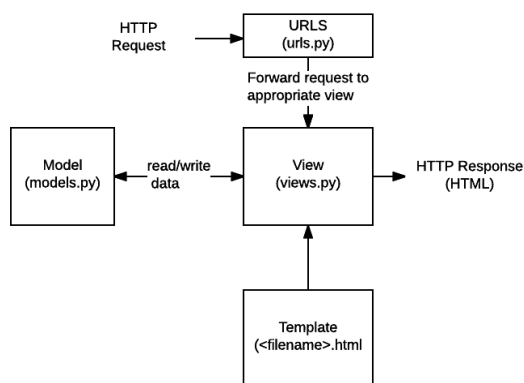


Figure 3: System Architecture of Django

2.4.3. Database Performance

Django’s database layer provides various ways get the best performance from their databases. The database optimization documentation gathers links together to the relevant documentation and adds various tips that outline to take the steps to make database usage optimized.

2.4.4. API Support

Many frameworks allow you to easily build APIs for applications, but only Django REST framework. It’s convenient in many ways and advantages are:

- 1) Its Web browsable API has huge usability.
- 2) Authentication policies include packages for OAuth1 and OAuth2.
- 3) Serialization supports both ORM and non-ORM data sources.
- 4) It has extensive documentation and great community support.
- 5) It’s employed by internationally recognized companies including Mozilla, Red Hat, Heroku, and Eventbrite.

2.5 ASP.NET

It is set of web development produce by Microsoft. It is used to built dynamic web sites, web applications, XML-based applications using Virtual Basics, .Net, C#.NET is a general purpose development platform for web, mobile, desktop, gaming, and IoT for Windows, macOS, Linux, Android, and more. It provide tight integration with

visual studio, stable code but it is too expensive to deploy and maintain. Tools Integrate with asp.net are: C#, Microsoft Azure, Sentry, Datadog.

2.5.1. System Architecture

ASP.NET works on three tier architecture. This architecture increase application performance, scalability, flexibility, and code reuse. In three tier architecture, applications are divided into three major areas of functionality:

- 1) **The Data Layer:** All the code related to database is written during this layer. It contains methods to connect with database. It performs insert, update, delete, select data operations according to the SQL query. This layer manages the data storage and retrieval.
- 2) **The Business Layer:** Business layer or Application layer acts between Application layer and Data Access Layer. This layer manage business rules and logic. It communicate with data layer.

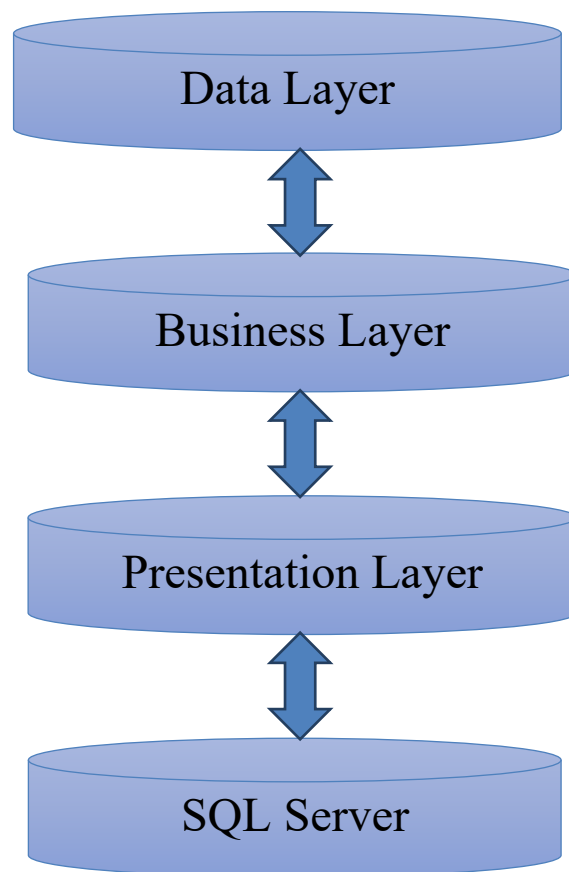


Figure 4: .Net architecture

3) The Presentation Layer:

The presentation layer acts as an interface between user and application or website. It contains .aspx pages where data is presented to the user or user enters the information. It communicates with Business layer

2.5.2. Performance

.NET has a facility of Garbage Collector which can handle large size of data. Performance can be increase by avoiding use of sessions or application variable, Avoid bulk data store on client side etc.

2.5.3. API Support

The ASP.NET Web API is an extensible framework for building HTTP based services that can be accessed in different applications on different platforms such as web, windows, mobile etc. It only supports HTTP protocol. It supports different forms of data response. ASP.NET Web API framework includes new Http Client to communicate with Web API server.

3. CONCLUSION

The choice of one web development technology is always be hard for a web application developer. The developer has to consider many factors for useful, attractive and efficient development process. The research paper compares the prominent and popular web development technologies; to get the best technology around for web application development. We compared various research done previously on ASP.net, Codeigniter, Angular JS, Cake PHP, Django. On the basis of this literature survey we compared these tech

	System Architecture	ORM technique	Performance	API support
Cake PHP	MVC based RUBY framework	Yes	Faster for small project, No data binding process	Implementation is difficult
CodeIgniter	MVC based PHP framework	No	Enhanced stability and support	No built in support
Angular JS	Component-oriented +(MVC and MVVM	Yes	Best for single page web application	Fully extensible and supports
Django	MTV based python framework	Yes	Great performance	Built in API support with great community support
ASP.net	Three tier architecture	Yes	High performance	Various and latest API support

Figure 4: .net framework layers

nologies based on System architecture, ORM technique, performance, API support. The comparative study is represented in table.

Table 1: Comparative study results

4. REFERENCES

[1] Nourie D. (November, 2006), Java Technologies for Web Applications, Oracle Technology Network, Retrieved from <http://www.oracle.com/technetwork/articles/javase/ssLINK/142892#nourie>

[2] Swales D., Sewry D. and Terzoli A (2003), A Performance Comparison of Web Development Technologies to Distribute Multimedia across an Intranet, Retrieved from <http://www.satnac.org.za/proceedings/2003/bband/bband2/703 - Swales.pdf>

[3] Lavanya, R., Ramachandran, V., & Mustafa, J. (2010), A Comparative Study on Internet Application Development Tools, International Journal of Engineering Science and Technology, 2 (10), 5452-5456.

[4] <https://www.diva-portal.org/smash/get/diva2:831439/FULLTEXT01.pdf>

[5] <https://www.valuecoders.com/blog/technology-and-apps/codeigniter-vs-cakephp-vs-yii-vs-laravel/>

[6] <https://ieeexplore.ieee.org/document/7550838>