Web-based frameworks

Chapter 4

Web-Technology-2 Note made by Lecturer
Nabraj Koirala, CCT College Butwal

content management system (CMS)

- Content management system (CMS), is a set of tools that provides an organization with a way to manage digital information on a website through creating and maintaining content without prior knowledge of web programming or markup languages.
- Example are WordPress, Drupal, and Joomla

PHP Frameworks

- A PHP Framework is a basic platform that allows us to develop web applications. PHP Framework, helps for stopping the need to produce repetitive code, and you'll be able to build applications rapidly (RAD). Without a PHP Framework in place, it gets much more difficult to produce applications since you'll have to repeatedly code a lot of PHP.

- PHP operates on the Model View Controller (MVC) fundamentals.
- **M**odel **V**iew **C**ontroller or **MVC** as it is popularly called, is a software design pattern for developing web applications. A Model View Controller pattern is made up of the following three parts −
- **Model** − The lowest level of the pattern which is responsible for maintaining data.
- **View** − This is responsible for displaying all or a portion of the data to the user.
- **Controller** − Software Code that controls the interactions between the Model and View.

- **What to Look for When Choosing Your Frameworks?**

a) Database Support

- Database support is very important. For example, CodeIgniter framework supports MySQL, Oracle, and SQLite, while the Kohana framework doesn't support Oracle or SQLite. Depending on which database you prefer to use or choose for your project at hand.

b) Community Support

- Your framework should have a strong community, not just in terms of size but also in activity and helpfulness. Even if it's a small community, as long as you're able to find support, then that's a plus point.

c) Documentation Support

- You should also be tired of frameworks that don't have any documentation and absolutely no user guide. Make sure that your PHP Framework has good documentation that's kept up-to-date, and that the user guide its relatively easy to follow.

PHP framework:

1) Laravel
2) Symfony

Laravel

- Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.
- Laravel offers a rich set of functionalities which incorporates the basic features of PHP frameworks. Laravel has a very rich set of features which will boost the speed of web development.
- If you are familiar with Core PHP and Advanced PHP, Laravel will make your task easier. It saves a lot time if you are planning to develop a website from scratch. Moreover, a website built in Laravel is secure and prevents several web attacks.

Advantages of Laravel

Laravel offers you the following advantages, when you are designing a web application based on it −

- The web application becomes more scalable, owing to the Laravel framework.
- Considerable time is saved in designing the web application, since Laravel reuses the components from other framework in developing web application.
- It includes namespaces and interfaces, thus helps to organize and manage resources.

Features of Laravel

Laravel offers the following key features which makes it an ideal choice for designing web applications −

- **Modularity:**

  Laravel provides 20 built in libraries and modules which helps in enhancement of the application. Every module is integrated with Composer dependency manager which eases updates.

- **Testability:**

  Laravel helps in testing through various test cases. This feature helps in maintaining the code as per the requirements.

- **Routing**

  Laravel provides a flexible approach to the user to define routes in the web application. Routing helps to scale the application in a better way and increases its performance.

- **Configuration Management**

  A web application designed in Laravel will be running on different environments. Laravel provides a consistent approach to handle the configuration in an efficient way.

- **Query Builder and ORM**

  Laravel incorporates a query builder which helps in querying databases using various simple methods. It provides **ORM** (Object Relational Mapper) and **ActiveRecord** implementation called Eloquent.

- **Schema Builder**

  Schema Builder maintains the database definitions and schema in PHP code.
- **Template Engine**

  Laravel uses the **Blade Template** engine, a lightweight template language used to design hierarchical blocks and layouts that include dynamic content.
- **E-mail**

  Laravel includes a **mail** class which helps in sending mail with rich content and attachments from the web application.
- **Authentication**

  User authentication is a common feature in web applications. Laravel eases designing authentication as it includes features such as **register, forgot password.**

- **Redis**

  Laravel uses **Redis** to connect to an existing session. Redis interacts with session directly.

- **Queues**

  Laravel includes queue services like emailing large number of users. These queues help in completing tasks in an easier manner without waiting for the previous task to be completed.

- **Event and Command Bus**

  Laravel includes **Command Bus** which helps in executing commands and dispatch events in a simple way.

2) Symfony

- Symfony is an open-source PHP web application framework, designed for developers who need a simple and elegant toolkit to create full-featured web applications.
- Symfony is an open-source MVC framework for rapidly developing modern web applications. It contains a set of reusable PHP components. You can use any Symfony components in applications, independently from the framework.

- The Symfony framework consists of several components, such as the HttpFoundation component that understands HTTP and offers a nice request and response object used by the other components. Others are merely helper components, such as the Validator, that helps to validate data. Kernel component is the heart of the system. Kernel is basically the ‘main class’ that manages the environment and has the responsibility of handling a http request.
- Symfony’s well-organized structure, clean code, and good programming practices make web development easier. Symfony is very flexible, used to build micro-sites and handle enterprise applications with billions of connections.

Symfony Framework - Features

- Model-View-Controller based system
- High-performance PHP framework
- Flexible URI routing
- Code reusable and easier to maintain
- Session management
- Error logging
- Full-featured database classes with support for several platforms
- Supports a huge and active community
- Set of decoupled and reusable components
- Standardization and interoperability of applications
- Security against cross-site request forgery and other attacks
- Twig(branch) template engine

- Symfony offers a lot of flexibility to developers. It has great features for debugging, code readability, and developing extensible programs.
- Symfony is a full-stack web framework; it is a very effective tool for creating web applications. Numerous companies offer Symfony services to clients.

Following are some of the benefits that you get by using the Symfony Framework:

- **Microframework −** Symfony can be used to develop a specific functionality.
- Reduces development time overhead.
- Extremely mature templating engine and quickly delivers content to the users.
- **Compatible and extensible −** Programmers can easily extend all framework classes.

Java frameworks

1) **Struts framework**
2) **Spring framework**

Struts:

- **Struts** is a popular and mature web application framework based on the MVC design pattern.
- The Webwork framework initially started with Struts framework as the basis and its goal was to offer an enhanced and improved framework built on Struts to make web development easier for the developers.

Struts Framework Features

- **POJO(**Plain Old Java Object**) Forms and POJO Actions** − Struts has done away with the Action Forms that were an integral part of the Struts framework. With Struts, you can use any POJO to receive the form input. Similarly, you can now see any POJO as an Action class.
- **Tag Support** − Struts has improved the form tags and the new tags which allow the developers to write less code.
- **AJAX(**Asynchronous JavaScript and XML**) Support** − Struts has integrated AJAX support into the product by creating AJAX tags.
- **Easy Integration** − Integration with other frameworks like Spring, Tiles etc is now easier with a variety of integration available with Struts.

- **Template Support −** Support for generating views using templates.
- **Plugin Support −** The core Struts behavior can be enhanced and augmented by the use of plugins.
- **Profiling −** Struts offers integrated profiling to debug and profile the application. In addition to this, Struts also offers integrated debugging with the help of built in debugging tools.
- **Easy to Modify Tags −** Tag markups in Struts can be tweaked using Freemarker templates. This does not require JSP or java knowledge. Basic HTML, XML and CSS knowledge is enough to modify the tags.
- **Promote Less configuration −** Struts promotes less configuration with the help of using default values for various settings. You don't have to configure something unless it deviates from the default settings set by Struts2.
- **View Technologies −** Struts has a great support for multiple view options (JSP, Freemarker etc)

Pros:

- Free And Open-Source
- Fast Development
- Easy To Test New Code

Cons:

- Many Rules
- Complex Framework
- Not Very Flexible

2) Spring

- Spring is the most popular application development framework for enterprise Java. Millions of developers around the world use Spring Framework to create high performing, easily testable, and reusable code.
- Spring framework is an open source Java platform. It was initially written by Rod Johnson and was first released under the Apache 2.0 license in June 2003.
- Spring is lightweight when it comes to size and transparency. The basic version of Spring framework is around 2MB.

- The core features of the Spring Framework can be used in developing any Java application, but there are extensions for building web applications on top of the Java EE platform. Spring framework targets to make J2EE development easier to use and promotes good programming practices by enabling a POJO-based programming model.

Benefits of Using the Spring Framework

- Following is the list of few of the great benefits of using Spring Framework −
- Spring enables developers to develop enterprise-class applications using POJOs. The benefit of using only POJOs is that you do not need an EJB(Enterprise Java Bean) container product such as an application server but you have the option of using only a robust servlet container such as Tomcat or some commercial product.
- Spring is organized in a modular fashion. Even though the number of packages and classes are substantial, you have to worry only about the ones you need and ignore the rest.
- Spring does not reinvent the wheel, instead it truly makes use of some of the existing technologies like several ORM frameworks, logging frameworks, JEE(Java Enterprise Edition), Quartz and JDK timers, and other view technologies.

- Testing an application written with Spring is simple because environment-dependent code is moved into this framework. Furthermore, by using JavaBeanstyle POJOs, it becomes easier to use dependency injection for injecting test data.
- Spring's web framework is a well-designed web MVC framework, which provides a great alternative to web frameworks such as Struts or other over-engineered or less popular web frameworks.
- Spring provides a convenient API to translate technology-specific exceptions (thrown by JDBC, Hibernate, for example) into consistent, unchecked exceptions.
- Lightweight IoC(Inversion **of** Control) containers tend to be lightweight, especially when compared to EJB containers, for example. This is beneficial for developing and deploying applications on computers with limited memory and CPU resources.
- Spring provides a consistent transaction management interface that can scale down to a local transaction (using a single database, for example) and scale up to global transactions.

Pros:

- Excellent Documentation
- Extremely Helpful Community
- Broad And Expansive Toolkit For Any Project You Might Have
- Enables You To Write Clean And Accessible Code

Cons:

- Steep Learning Curve
- Requires Quite A Bit Of Pre-Existing Knowledge Before Using