Yasir Arafath

Content Management Systems: An overview between WordPress and Drupal

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Bachelor's Thesis

8 December 2021

Abstract

Author:	Yasir Arafath
Title:	Content Management Systems: An overview between WordPress and Drupal
Number of Pages:	35 pages
Date:	8 December 2021
Degree:	Bachelor of Engineering
Degree Programme:	Information Technology
Professional Major:	Software Engineering
Supervisor:	Janne Salonen, Head of the department

This study presents a bachelor's thesis on the topic of Web Content Management Systems: An overview between WordPress and Drupal. The aim of this study is to get general idea about Content Management Systems.

This thesis includes a theoretical background of Content Management Systems, its history, types, and architecture. Later it describes WordPress and Drupal in details in terms of inception, how it works, and architecture.

The final part of this thesis gives a comparison between WordPress and Drupal in the areas of market share, security, installation, customization, ease of use, content management and performance.

Keywords: CMS, WordPress, Drupal

Contents

List of Abbreviations

1 Introduction 1

2 Theoretical Background of Content Management Systems 2

2.1 History of Content Management Systems 2
2.2 Types of Content Management System 4
2.2.1 Web Content Management System 4
2.2.2 Digital Asset Management System 4
2.2.3 Document Management System 4
2.2.4 Enterprise Content Management System 5
2.2.5 Component Content Management System 5
2.3 Architecture of Content Management System 5
2.3.1 Coupled Content Management System Architecture 5
2.3.2 Decoupled Content Management System Architecture 6
2.3.3 Headless Content Management System Architecture 7
2.3.4 Hybrid Content Management System Architecture 8

3 WordPress 10

3.1 History of WordPress 11
3.2 How WordPress works 13
3.3 WordPress Themes 15
3.4 WordPress Architecture 16

4 Drupal 20

4.1 History of Drupal 21
4.2 How Drupal works 24
4.3 Drupal architecture 28

5 Comparison between WordPress and Drupal 31

5.1 Market share 31
5.2 Security 31
5.3 Installation 32
5.4 Customization 33

5.5 Ease of use 33
5.6 Content Management and Performance 33
6 Conclusion 35
References 36

List of Abbreviations

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
ASP	Active Server Pages
CDA	Content Delivery Application
CMA	Content Management Application
CMS	Content Management System
DOM	Document Object Model
GNU	GNU’s Not Unix
HTML	Hyper-Text Markup Language
JSP	Java Server Pages
PHP	Personal Home Page
SGML	Standard Generalized Markup Language
SSI	Server Side Includes
UI	User Interface
XML	Extensible Markup Language

1 Introduction

Content Management System (CMS) refers to the software application or related programs to create digital contents. It helps user to manage large amounts of web-based information and modify the content on a website without specialized technical knowledge. [1.]

In other words, CMS is a tool that helps to build a website without proper knowledge of coding. Even, in some cases it is possible to build a website with zero knowledge of coding. For example, CMS can be able to handle the basic infrastructure of a website such as creating web pages, storing images and other necessary functions to run the website successfully. [2.]

CMS usually server based which allows multiusers to create new content and edit existing content in the same repository. The repository can locate in the same server or entirely different server. CMS consists different types of parts, for instance, editing interface, repository, publishing mechanism et cetera. [3.]

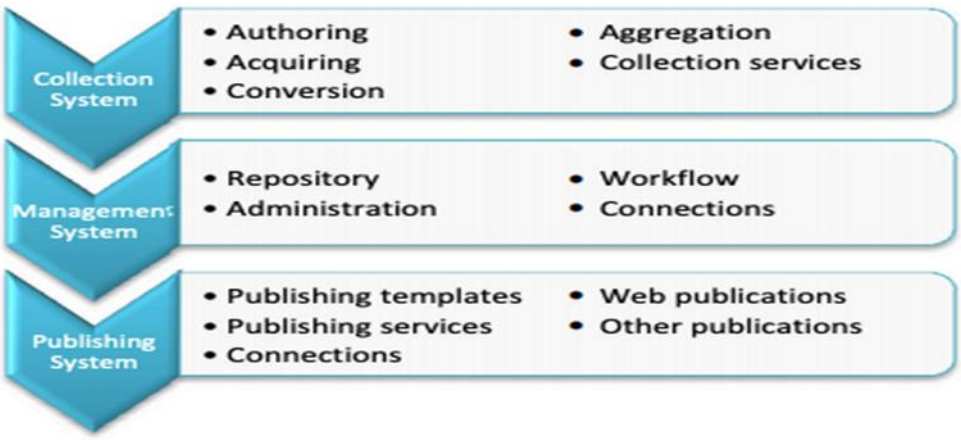

Figure 1. Overview of Content Management System. [4.]

Generally, CMS has two main components. These are: Content Management Application (CMA) and Content Delivery Application (CDA). CMA works as the front-end user interface which allows user to add, modify or remove the content

and CDA helps to update the content on the website. CMS mainly divided into three functions. These are: Collection System, Management System and Publishing System. [5.]

2 Theoretical Background of Content Management Systems

In this part author will discuss the theoretical background of Content Management Systems in terms of history, types, and architecture.

2.1 History of Content Management Systems

The history of CMS starts with the proposal of internet-based hypertext system. Tim Berners-Lee in 1990 developed Hypertext Markup Language (HTML) for the browser and the server. HTML came from Standard Generalized Markup Language (SGML) which was created by Charles F. Goldfarb, Ray Lorie and Ed Mosher in 1970s at IBM. Initially, websites were very simple, able to handle only HTML text files. Later, in 1993 Mosaic browsers starts to support image which can be appeared with text. This was the beginning of static brochure like web sites which can be able to share company and product information. In 1990s it was possible to manage the content in a website with the support of Server Side Includes (SSI). Server Side Includes helped to separate the site menu and footer from the main content. Common Gateway Interface was introduced around the same time to create interactive forms. [6.]

Initial CMS used Server-Side Scripting technologies to generate content in the web browser. Later, in the late 1990s some programing languages like Personal Home Page (PHP), Active Server Pages (ASP) and Java Server Pages (JSP)

eased the CMS in such a way that it is becoming simple to generate dynamical web pages. This was the beginning of CMS's popularity. [7.]

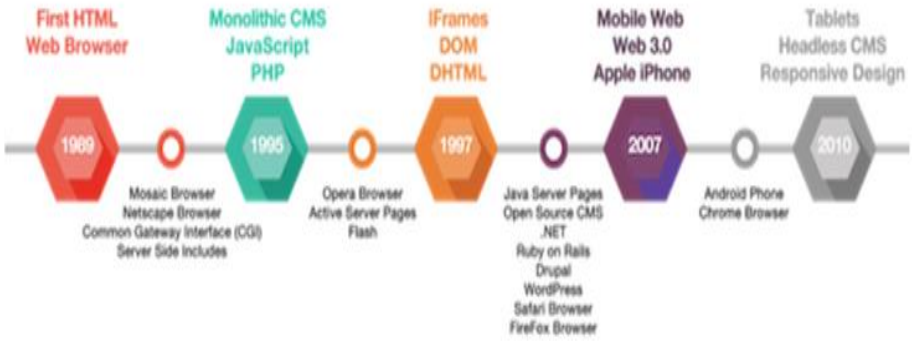

Figure 2. Milestone for Web Content management [6.]

The introduction of Document Object Model (DOM) in 1997 was phenomenon because DOM was able to define the logical structure of the web page and can control the document programmatically. For instance, DOM can access and manipulate the entire HTML page like the body or a division of the page. Later introduction of Asynchronous JavaScript and XML also known as AJAX gives the opportunity to the Developers to update the content of the Web page without reloading the page. As a result, many organizations create their own customized CMS by using available technology between 1997 and 2002. During the period of early to mid-2000s CMS becoming more professional and specialized to fulfill the business needs. This is the period when CMS for example, WordPress gets popularity. [7.]

2.2 Types of Content Management System

CMS offers significant benefits to the organizations. To take an example, Companies can focus on their core activities instead of wasting their valuable resource on the web page, which is easily manageable by CMS. At present, there are several options available from which companies can choose CMS based on their needs. These are:

2.2.1 Web Content Management System

A Web Content Management System offers organizations to manage their digital presence in the form of a web page without technical skills of Web programming. In this system content editor use predefined templates to create the structure of the website. The benefits of this systems are Personalization, Automation and Scalability. [8.]

2.2.2 Digital Asset Management System

This Content Management System is specially designed for Photography, Graphic Design and Film companies where visualization is the core components of the business. In this system companies can share their digital assets through cloud-based system. The benefits of Digital Asset Management Systems are: Centralized Repository, Effective Brand Management and Digital Publishing. [8.]

2.2.3 Document Management System

Document Management System focus on the content of a file level instead of the broader details. This system offers automated solution for sharing business documents in the cloud. Benefits of this system are Eco friendly, Security and Mobile solution. [8.]

2.2.4 Enterprise Content Management System

Enterprise Content Management System provides diverse solution to maintain the content for instance, email, documents, instant message et cetera. This is very useful for large organizations to update their contents and keep connection with customers. The benefits of this system are Flexibility, Efficiency and Cost Reduction. [8.]

2.2.5 Component Content Management System

Component Content Management System offers deep level solution to manage the content which is different from standard level solution. In this system, content was managed in micro levels for example, words, photos, paragraphs and later stored in the central repository. This ensures the maximization of the reusability of the content where it needs to store only once. The benefits of this system are Traceability, Reusability, Team Collaboration and Single Sourcing. [8.]

2.3 Architecture of Content Management System

Based on Architecture, CMS can be divided into four types. These are:
1. Coupled Content Management System
2. Decoupled Content Management System
3. Headless Content Management System
4. Hybrid Content Management System

2.3.1 Coupled Content Management System Architecture

In Coupled architecture backend and frontend closely work together to provide general functionality. This is also known as “Traditional Content Management System”. The elements of backend such as database, interface, publishing engine are directly linked and integrated whereas, the frontend mainly used for displaying purpose. [9.]

Figure 3. Coupled CMS Architecture.

The main advantages of this kind of architecture are accessibility and strong integration which allows users to create a webpage as a medium of blog. On the other hand, the biggest disadvantages of this architecture are common design, limited development environment and database restriction. [9.]

2.3.2 Decoupled Content Management System Architecture

Decoupled architecture is more complex compared to Coupled architecture. In this architecture frontend and backend works separately by using existing API. Here backend has own set of rules to host the database where all the contents are stored, and frontend is responsible for providing unique interface by following own schema. [9.]

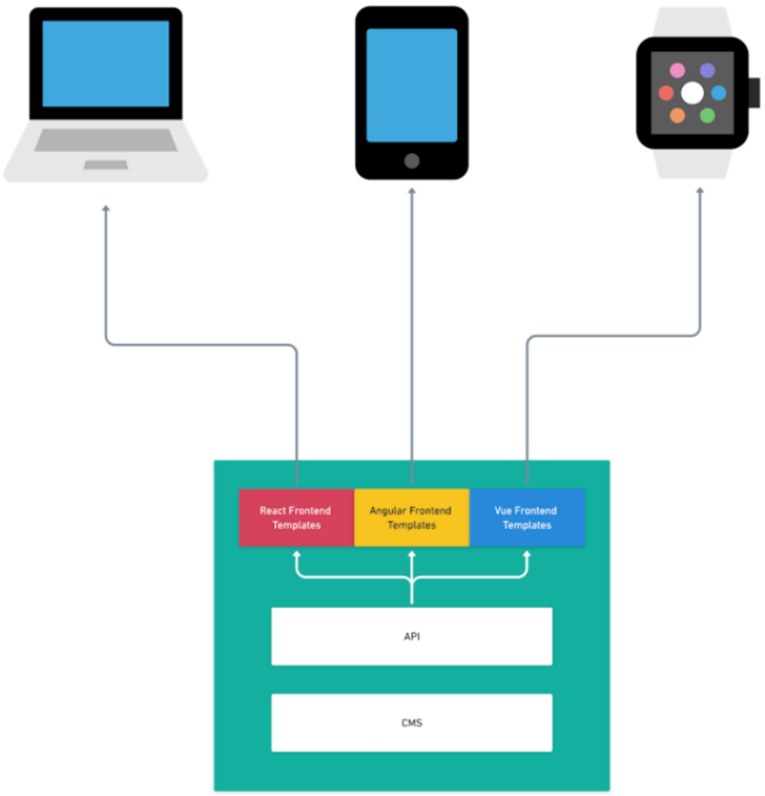

Figure 4. Decoupled CMS Architecture.

The positive side of this kind of architecture are separation of content creation and publishing, isolated code complexity and multichannel publishing. The negative side of this architecture are complexity of the system and need to have the knowledge about publishing environment. [9.]

2.3.3 Headless Content Management System Architecture

Headless Architecture is different from Coupled Architecture; however, in a way it is close to Decoupled Architecture. This Architecture is called ‘Headless’ because it does not have any dedicated frontend panel to represent the content and developers can create their own version of frontend. Here frontend and backend are separated from each other, and communication achieved through API by using various channel. [9.]

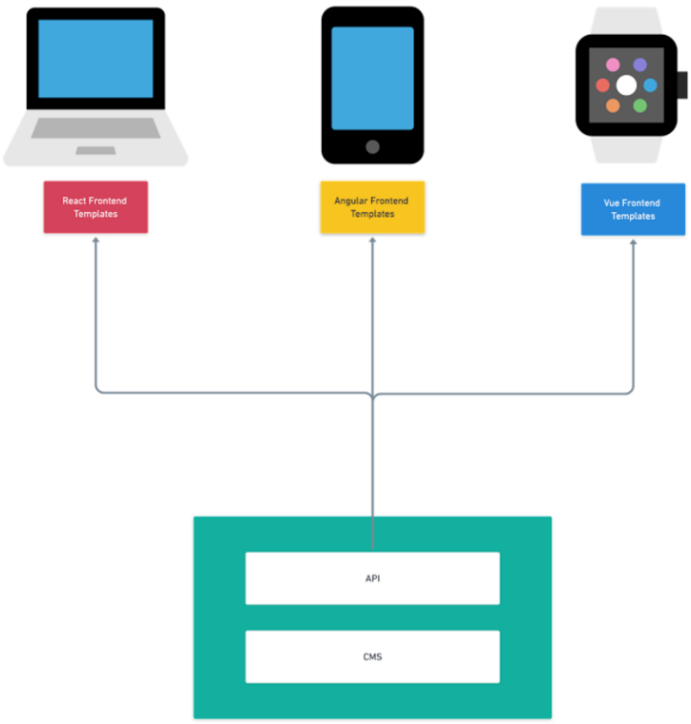

Figure 5. Headless CMS Architecture.

The advantages of this Architecture are fast delivery content, flexibility for Developers. On the other hand, the disadvantages are no preview method to publish the content easily and dedicated customization needed for publishing. [9.]

2.3.4 Hybrid Content Management System Architecture

Hybrid Architecture was built on top of Headless architecture and follow the same pattern of Decouples architecture. The difference between Hybrid architecture and Headless architecture is “channel silos” which refers to the cross-communication tool between endpoints. In Hybrid architecture, backend works like Headless architecture and frontend functions like Coupled architecture. [9.]

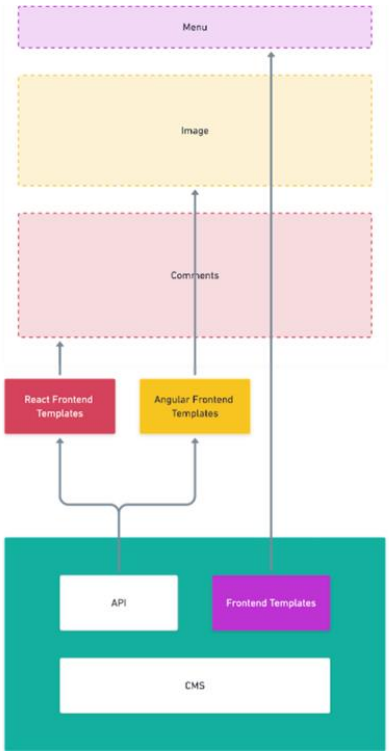

Figure 6. Hybrid CMS Architecture.

The benefits of this architecture are flexibility in editing and provides more control over the app integration whereas, the drawbacks are learning curves are difficult and it requires considerable amount of investment in terms of time and money. [9.]

3 WordPress

WordPress is an open-source Content Management System (CMS) based on PHP and MYSQL. WordPress allows its users to build websites and publish online easily. It registered under GNU Public License. This helps WordPress users to download and use it freely. [10.]

Figure 7. WordPress.

Initially, WordPress was built as a simple blog platform. Due to its growing popularity WordPress transforms into a feature rich Content Management System (CMS). The beauty of WordPress is that it allows anyone to create professional website without technical knowledge. [10.]

There are two versions of WordPress. One is “wordpress.org” and another one is “wordpress.com”. “wordpress.org” which is a self-hosted version and free, is very popular due its flexibility and control over the content. On the other hand, “wordpress.com” which is not a self-hosted version and not free, host the publishing service on behalf of the users. [10.]

3.1 History of WordPress

The history of WordPress started in 2002 by a college student named Matt Mullenweg. The initial version of WordPress is developed based on existing blogging software named b2/cafelog which was discontinued by the creator. The main reasons were personal matter and lack of project and community leader. [11.]

In 2003 Matt Mullenweg and Mike Little create the first version of WordPress with the significant upgrades. It includes admin interface, new templates, and XHTML 1.1 compatible templates. [12.]

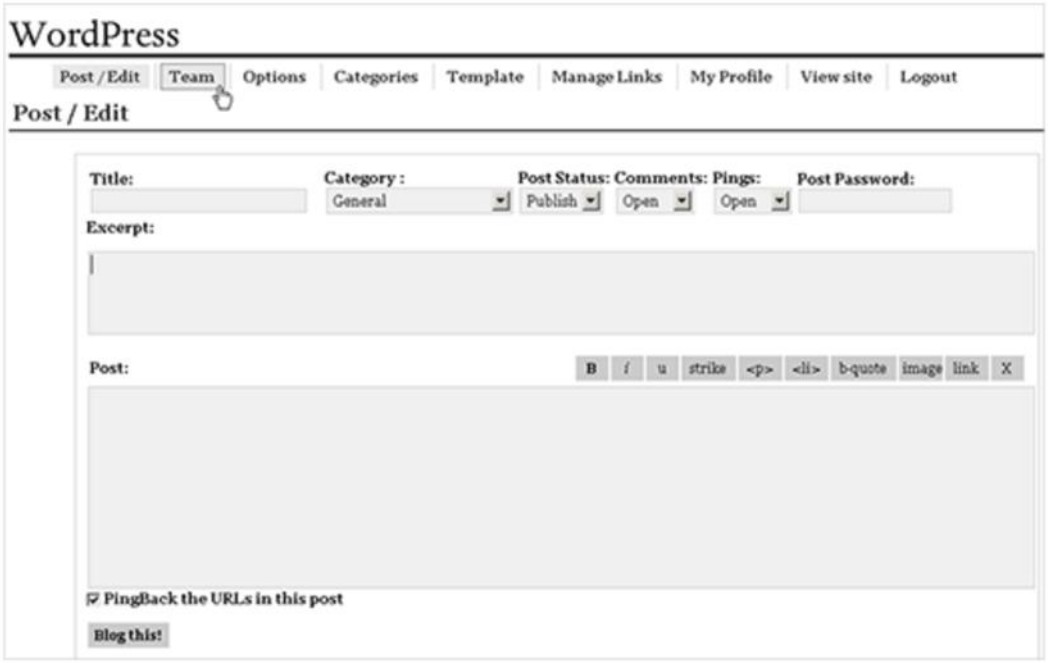

Figure 8. WordPress 1.1 post editor. [12.]

The second version of WordPress came out in 2004 with plugin architecture. The benefit of this architecture is that it helps users and developers to write their own plugins. This version also offers its users mature, stable and flexible platform. As a result, the number of WordPress users rise significantly. [12.]

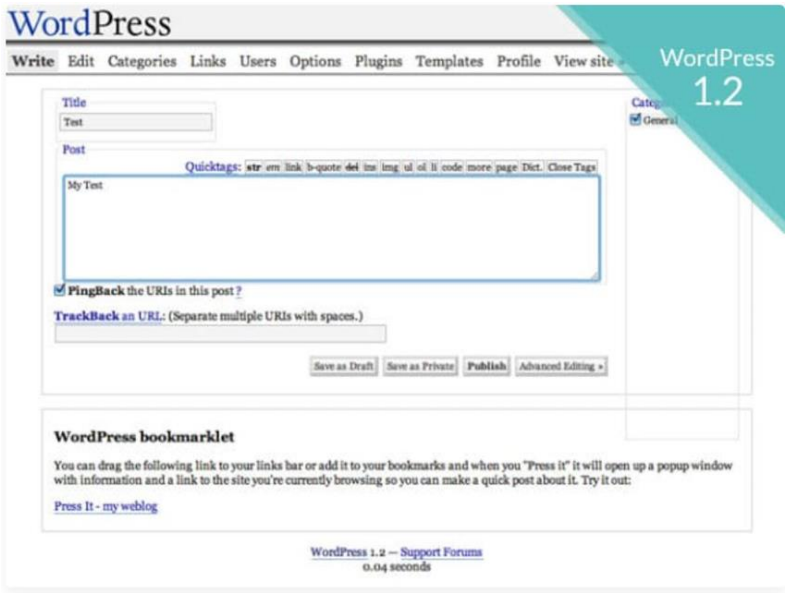

Figure 9. WordPress 1.2. [11.]

WordPress version 1.5 came in February 2005. It added some important features such as comment moderation tools, new default theme named as Kubrick and Theme system. [12.]

WordPress 2.0 came into effect in December 2005 with new admin dashboard. This admin dashboard is totally new from its previous version. It is using JavaScript and DHTML to give new User Interface (UI) to its users. [12.]

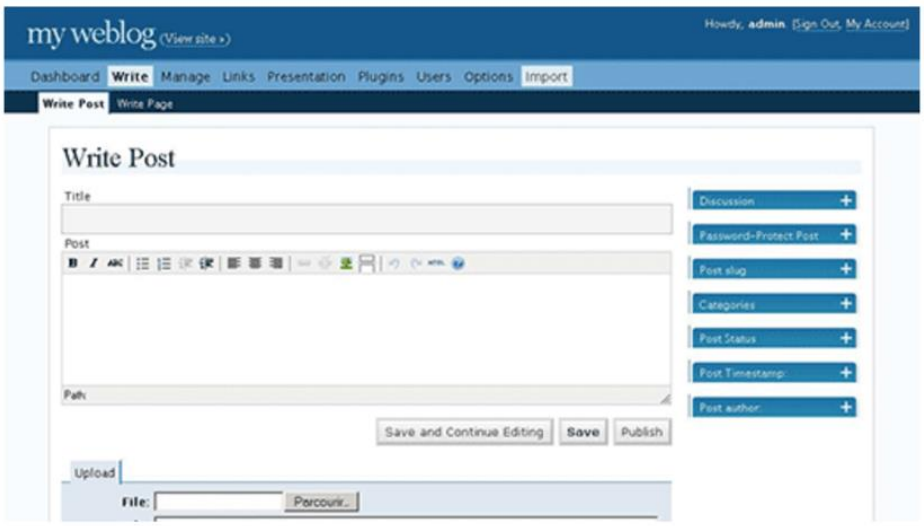

Figure 10. WordPress 2. [12.]

WordPress 3.0 came out in June 2010. To become Content Management System (CMS), in this version WordPress introduced some features such as Custom Post types, upgraded Custom Taxonomies, Custom backgrounds et cetera. [12.]

WordPress 4.0 was released in 2014 with updated writing and management experience. As a result, media management become easier. This version also improved search technic, metrics, and visual experience. [11.]

WordPress 5.0 came in 2018 with new block-based editing experience. The codename of this project is Gutenberg. With this version WordPress offers easily full site editing solutions. [12.]

3.2 How WordPress works

WordPress works in a different way compared to traditional website. In traditional website usually, a web designer writes a bunch of codes and keeps them in a folder of the web server. Therefore, when a visitor visits the address or request the web page then the web server respond or send the codes to the browser. [13.]

On the other hand, to create a web page, a user needs to provide raw data such as text, picture to WordPress. Later, when a visitor visits the site WordPress organized all the raw content based on category such as blog, article, or something else. [13.]

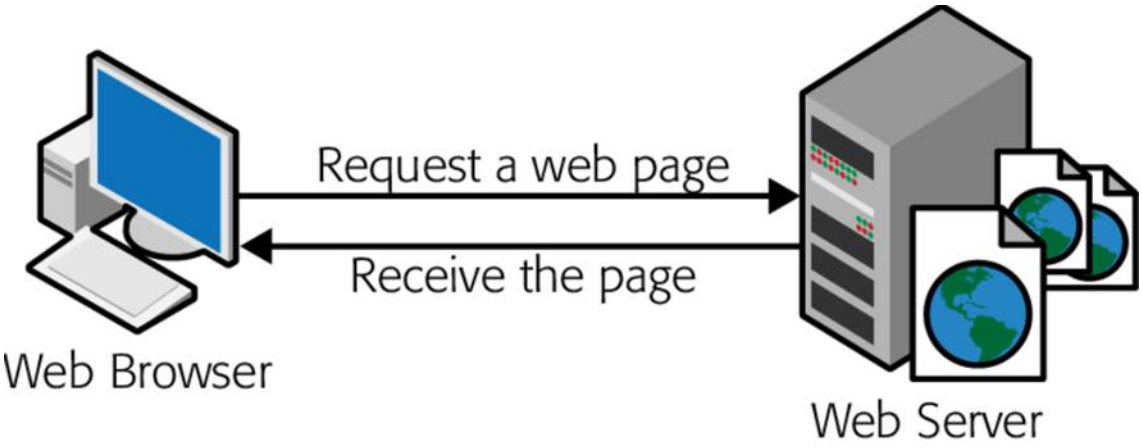

Figure 11. Request-Response cycle. [13.]

The benefit of this system is that it is arranged all the content in a meaningful way. For instance, when someone looking for a certain blog or post he or she can search it by using logical keyword such as based on month, topic, or certain keyword. In order to achieve the expected result WordPress combined programs and content on the web server. To take an example, when a visitor looking for a post by writing “Chocolate Cake” behind the scenes WordPress programmatically searching all the posts that matches the “Chocolate Cake” key words and later display the result on the visitor’s browser as a web page. [13.]

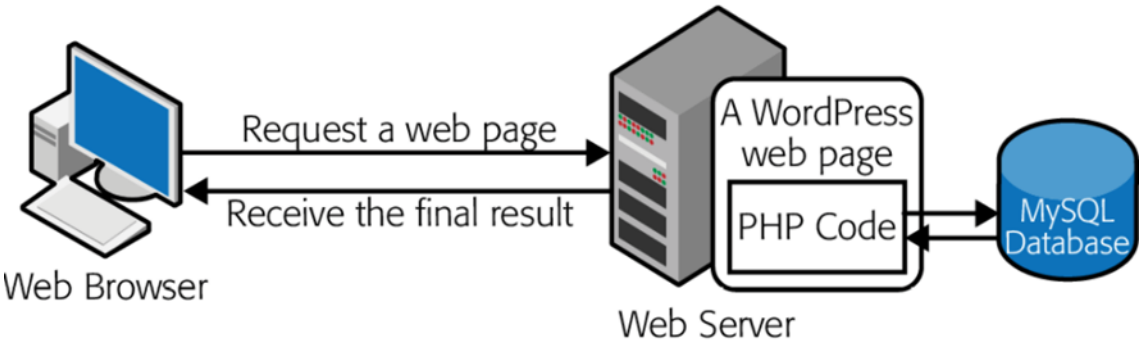

Figure 12. WordPress Request-Response cycle. [13.]

There are two key components that WordPress uses to display the results. These are: Database and Programming code. A database is a storage system that keeps all the relevant information for later use such as texts, pictures,

contents et cetera. WordPress uses MySQL as a database engine. The reason is that MySQL is open source and free. In case of programming code WordPress uses PHP. PHP helps WordPress to display and fetch all the information based on logic as a cohesive page. [13.]

3.3 WordPress Themes

WordPress has a feature called “Themes”. The idea of Theme is that it helps WordPress to separate the content from layout. The benefit of this Theme is that it gives flexibility to its users. By using Theme, it is possible to change the layout settings without changing the content. [13.]

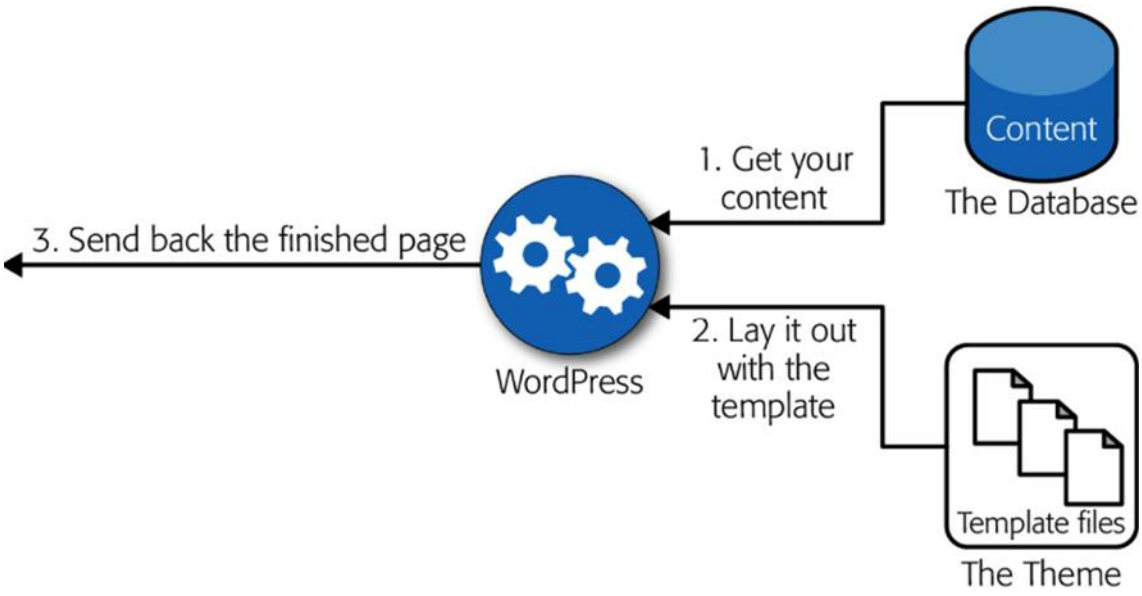

Figure 13. How WordPress Theme works. [13.]

For example, when a user visits a WordPress site, WordPress collects the content from the database and display the result based on the Theme which are written instructions in the template files. [13.]

3.4 WordPress Architecture

WordPress changed a lot from its inception. With the help of various contribution from the tech-savvy people and technology such as REST API at present WordPress evolved as a Headless Architecture. [13.]

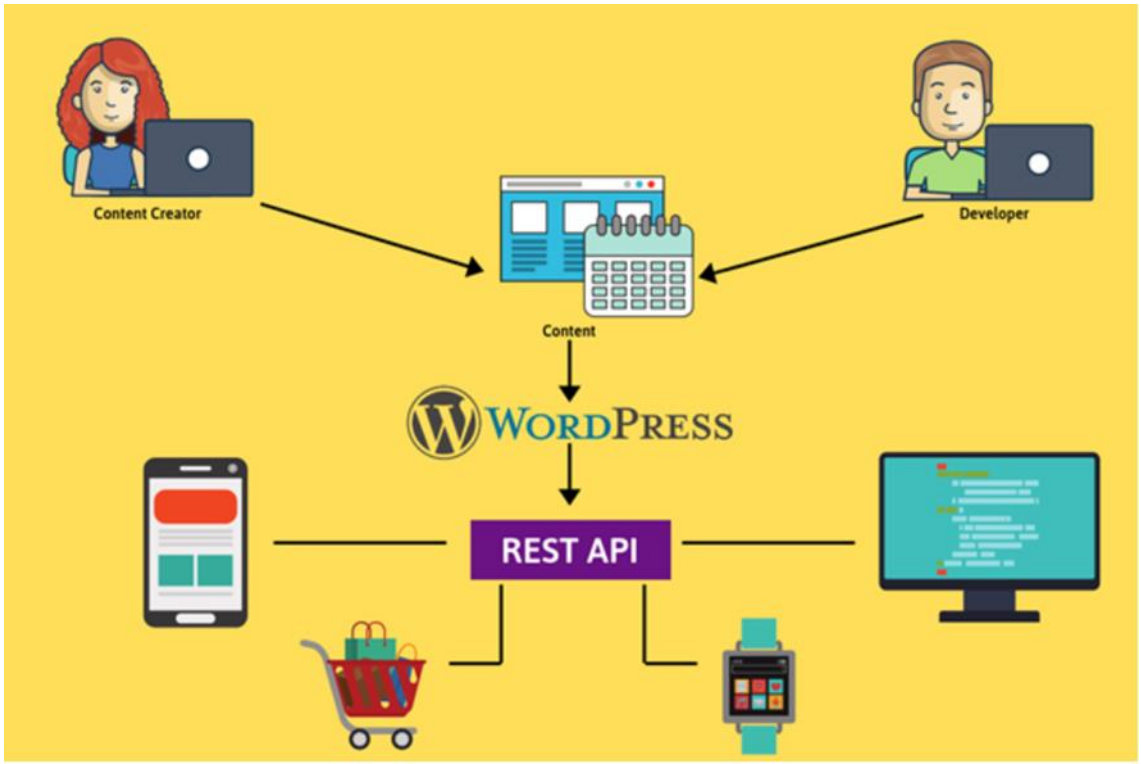

Figure 14. WordPress - ECommerce System Headless Architecture. [14.]

WordPress architecture follows Headless architecture, which is eventually a decoupled architecture. In decoupled architecture the architecture is divided into two parts. The first part consists of contents and relevant data structure and the second part consist of API or Web Service by the front-end part. In Headless architecture presentation layer or front-end part is not defined and developer can modify the front-end part based on needs. [14.]

By default, WordPress comes with a theme. A user can use the default theme or can modify the theme based on the needs. WordPress themes comprise with Template files, Stylesheet files, JavaScript files, Image files and other files such

as functions.php. These files harmonize to create a functional WordPress website. [15.]

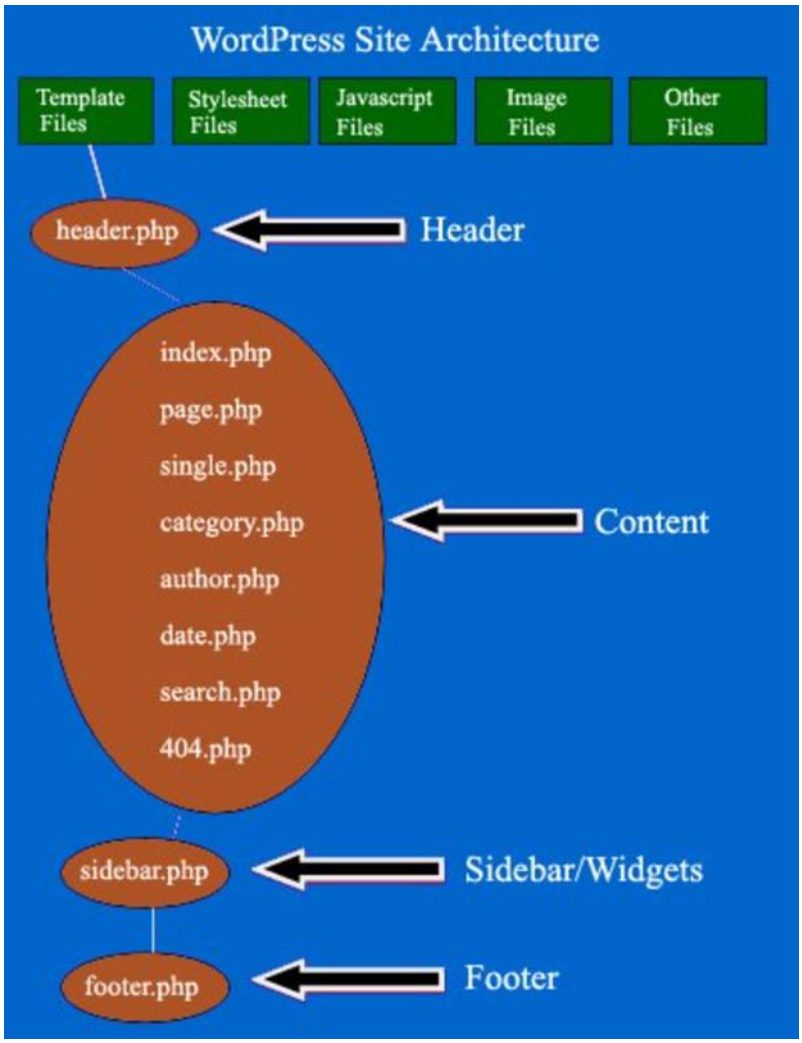

Figure 15. WordPress Site Architecture. [15.]

Template files: Template files are PHP files. These files are mixture of HTML, Template Tags and PHP codes. Template files creates web pages and allows customization in WordPress. For instance, header.php uses to create header part and comments.php uses to create comments. [15.]

When a visitor navigates a page on the WordPress website, based on request WordPress loads the template. If the template files presents in the theme then Template Hierarchy decides which template files to load . Later server parses the PHP file in the template and returns the HTML file in the browser. [16.]

Commonly used WordPress template files are:

header.php:

1. This file generates site's header and display navigation bar.
2. It contains document type, meta data and other information.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > header.php [15.]

index.php:

1. This is the main template file and it display blog post index.
2. It is possible to display the blog post index either on the front page or set as a different static page.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > index.php or Settings > Reading [15.]

home.php:

1. It is the default front page.
2. If home.php is not available, then WordPress use index.php to display blog post index.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > home.php [15.]

front-page.php:

1. This file works as a site front page.
2. It can be static page or blog post index.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > front-page.php [15.]

page.php:

1. This file shows page title and contents.
2. It is also used to see comment list and comment form.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > page.php [15.]

single.php:

1. This template is used to see a single post.
2. It can also display title, content, author's name et cetera from a post.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > single.php [15.]

category.php:

1. This template displays the posts based on category.
2. It is used when a request was placed based on category by the user.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > category.php [15.]

author.php:

1. This template is used when a visitor loads author page.
2. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > author.php [15.]

search.php:

1. This template shows visitor's search result.
2. It is possible to customize the search result by editing the file.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > search.php [15.]

404.php:

1. The 404-template file is used when WordPress fail to show the requested post, page or other contents.
2. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > 404.php [15.]

sidebar.php:

1. This template is used to generate sidebar.
2. It is possible to set the content of the sidebar.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > sidebar.php [15.]

footer.php:

1. This template is used to generate the footer.
2. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > footer.php [15.]

comments.php:

1. This file shows the comments.
2. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > comments.php [15.]

style.css:

1. This template is the mail stylesheet. It is essential for all themes.
2. It is possible to change the name, author, or URL by editing the file.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > style.css [15.]

functions.php:

1. This template is used to add custom feature to the theme.
2. If the file is not available in the theme, then it is possible to create the file by visiting wp-content/themes/yourThemeFolder by adding PHP functions.
3. To edit the file, need to visit WordPress Admin Panel > Appearance > Editor > functions.php [15.]

4 Drupal

Drupal is also a popular open-source content management systems to build dynamic web site. Drupal offers a wide range of features and services such as administration, publishing workflow, discussion capabilities, news aggregation, metadata functionalities and XML for content share. [17.]

Drupal is based on PHP language. This software is ready to use after download. It also provides web-based installer and add-on modules. Other features are content management, collaborative authoring, newsletters, podcasts, image galleries, peer to peer networking, file uploads et cetera. [18.]

Drupal is registered under the GNU General Public License, as a result it is free to use. It provides back-end framework to the web site. The standard release of Drupal also provides user account registration, maintenance, menu management, RSS feeds, taxonomy, page layout customization. Drupal supports modules which is equivalent to other Content Management System such as Joomla extensions and WordPress plugins. [19.]

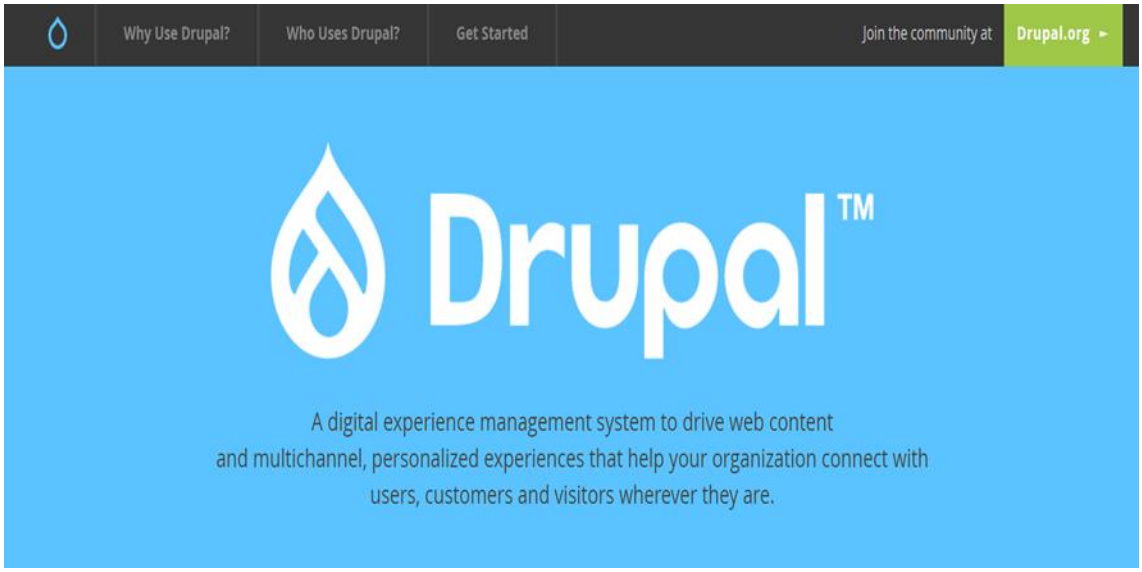

Figure 16. Drupal. [20.]

4.1 History of Drupal

In 2000, Dries Buytaert and Hans Snijder two students from University of Antwerp feel necessity of reliable internet connection. The reason is that at that time permanent internet connection was rare for Antwerp students. As a result, Dries Buytaert and Hans Snijder creates a wireless network in their dorms by sharing Hans Snijder's ADSL modem connection. However, they are still missing a place to talk to each other. Later Dries Buytaert creates a small news site which is a small content framework with the help of a web board. Now they can talk to each other, share their ideas and post notes about their status. [21.]

In the beginning, the software what Dries Buytaert and Hans Snijder created was without name. After Dries Buytaert graduation they decide to put the site in online, the reason was that to keep in touch and keep the communication. When searching for the domain name initially, Dries Buytaert thinks about "dorp.org". In Dutch "Dorp" means "village". While checking for the availability of "dorp.org" domain name by mistake he typed "drop" and found that mistake sounds better. In January 2001 Dries Buytaert release the software under "drop.org" domain name as a platform for web development experiment. The name "Drupal" came from Dutch word "druppel" which means "drop" in English. [21].

Drupal 1.0: January 2001

The original release of Drupal was based on Slash, modular CMS, and Scoop. Initially, there were 18 core modules. Each core module consists of PHP file which were a set of routines. At that time there was no menu router as a result, everything needs to go through PHP files. There were only seven hooks in the modules where the code needs to place. The admin hook only be able to manage by the administrators. In order to modify the database, need to import the SQL file. Themes were simple and flexible. It is also possible to create own themes. The firs version has full of many features such as story submissions, diary, accounts, comments, search, RDF headlines, calendar et cetera. [22.]

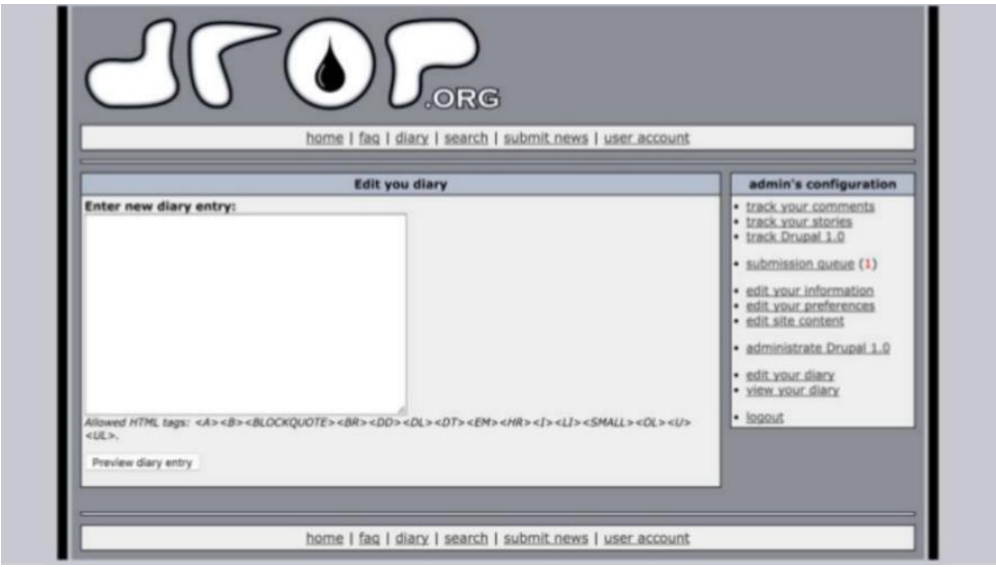

Figure 17. Drupal 1.0. [22.]

Drupal 2.0: March 2001

After the initial release Drupal 2.0 came in March 2001. In this version there is a feature called translation feature which allows developers to create or overwrite the site language, which was a major change because many people want to see their own languages. Other features are user ratings, sections for stories, user permission system, edit comment et cetera. [22.]

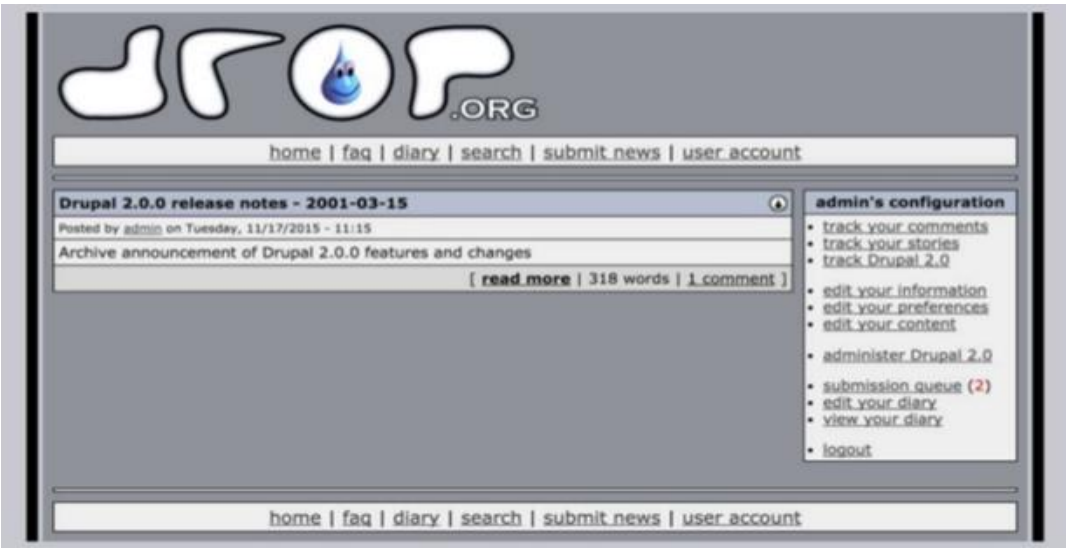

Figure 18. Drupal 2.0. [22.]

Drupal 3.0: September 2001

In this release everything based on nodes and nodes become the basic unit for all contents. For example, forum, diary, blog and only exception is comments. Later this node become the basis of mobile web. [22.]

Drupal 4.0: June 2002

The fourth addition of Drupal introduce taxonomy module which is useful to organizing tools. It also has an easy user interface and WYSIWYG editor which helps users to use Drupal without technical knowledge. As a result, Drupal community is rising and at that time almost 100 famous sites were built on Drupal platform. [23.]

Drupal 5.0: January 2007

In this version Drupal adopt jQuery which is JavaScript library to make web development easier. Other features are pre-created customized Drupal packages, updated CSS management, improved directory structure et cetera. [23.]

Drupal 6.0: February 2008

Drupal 6 introduced new menu system which was written from the beginning to make it easier to use. Other features are drag-and-drop, menu items, taxonomy vocabulary, security improvements and terms. It also introduced new language system to support right-to-left languages. [24.]

Drupal 7.0: January 2011

Drupal 7 uses many new features such as strong password security, file and image handling, usability improvement, new database system et cetera. As a result, it was very easy to use and successful because of power, speed, and security. [25.]

Drupal 8.0: November 2015

In this version Drupal emphasis in the front-end administration system. Major changes are happened in core functionality, and it also decrease dependency on external modules. Responsive image and Mobile responsiveness are also introduced for the first time. By using Symfony framework and Twig templates it is able to change configuration management and editing system under the hood. [25.]

Drupal 9.0: December 2020

Drupal 9 came into effect in June 2020 with relatively easy upgrade process. This version imposes more strict requirements on server and software versions. For example, Drupal 9 requires PHP version 7.3 and Apache 2.4.7 or later version. As a consequence, many modules that works on Drupal 7 or 8 will not work in Drupal 9. To overcome this problem Drupal 9 needs, rewrite the coding from the scratch or replacement with updated alternatives. [25.]

4.2 How Drupal works

In this part author is going to discuss about how Drupal works in terms of technological stack, core, modules, information flow and architecture.

Technology Stack: The aim of Drupal is to provide inexpensive web hosting facilities and scale up opportunities. As a result, Drupal needs to use popular web technologies which are tightly coupled. Drupal technology stack is shown in Figure 19.

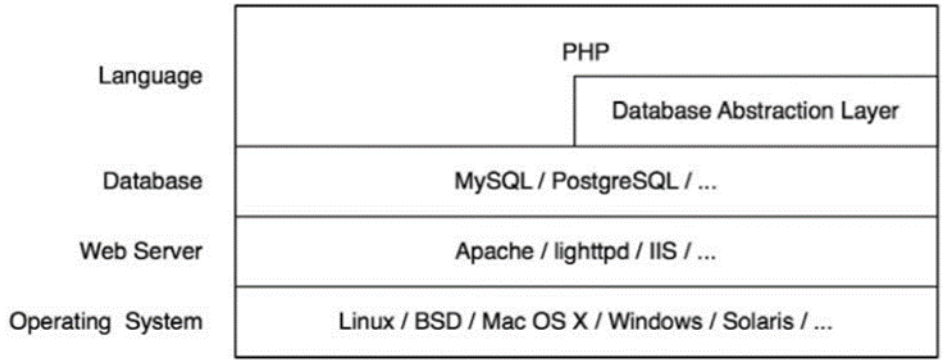

Figure 19. Drupal's technology stack. [26.]

The position of Operating System in the Drupal stack is at the bottom because Drupal is Operating System agnostic. Drupal can run any Operating System which support PHP. In case of, Web Server Drupal mainly used Apache. However, it is also possible to use other Web Servers such as IIS and lighttpd. In the next layer which is Database layer Drupal use those Databases which supports PHP, for example MySQL, PostgreSQL et cetera. Finally, as a programming language Drupal choose PHP. The reason is that PHP is easy to learn and there will be low learning curve for the contributors. [26.]

Core: Drupal core is built on a lightweight framework. It provides the basic functionality and supports other parts of the structure. It also provides modules such as Basic Content Management, User Management, Session Management, URL Aliasing, Localization, Taxonomy, Syndication, Templating, Logging. The core also provides aggregation, blogging, polls, forums et cetera which are the basic building block of modern web sites. [26.]

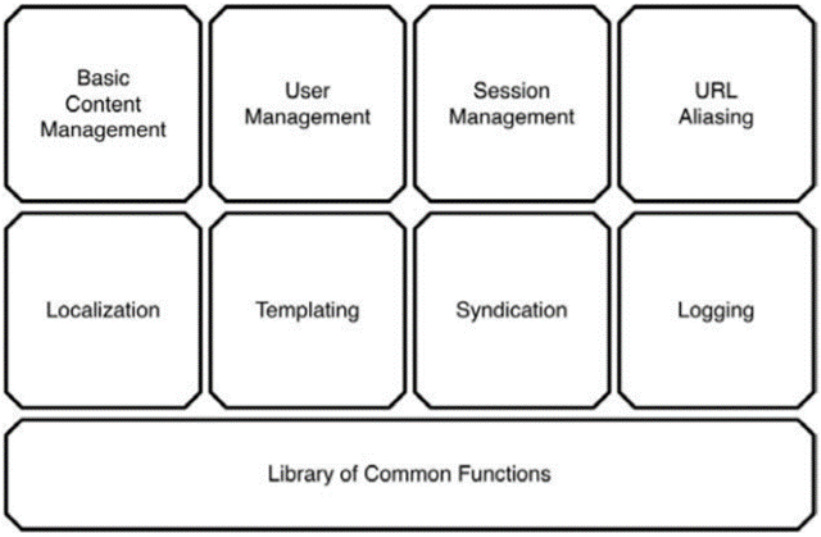

Figure 20. An overview of the Drupal core (not all core functionality is shown). [26.]

Modules: Drupal framework is based on modules. It is possible to have enable or disable functions through modules. In Drupal, modules have the ability to add new contents and functionalities for example, blog posts, files, recipes, e-mail notification, peer to peer publish, aggregation. [26.]

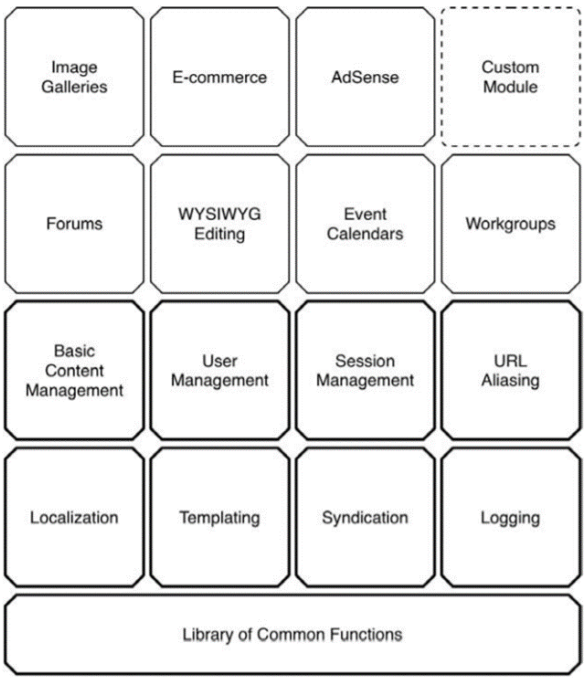

Figure 21. Enabling additional modules gives more functionality. [26.]

Information flow: In Drupal the direction of flow from bottom to top. The foundation of Drupal's information flow system is collection of Nodes. Nodes refer to data types such as title, text, comments, tags, author et cetera. In order to show any information on the site it needs to be input as data. The second layer of Drupal's information flow system is Modules. Modules use APIs for communication. Modules are event-drive which means Modules can 'listen' from hooks or events which are generated by the APIs. The third layer is Blocks and Menus. Blocks are useful to provide information from one module to another, for example through Blocks it is possible to show certain page based on geography or user. Menus help users to navigate on Drupal site by providing links to go other pages which are created in Drupal. The fourth layer of Drupal's information flow system is User Permissions. Based on permission users are allowed what to do or not. Through permission it is possible to define user roles on the site. The fifth layer of Drupal's information flow system is Template layer, which is made by XHTML, CSS, and PHP variables. This is the site theme. It is possible to assign the Template based on user permission. [27.]

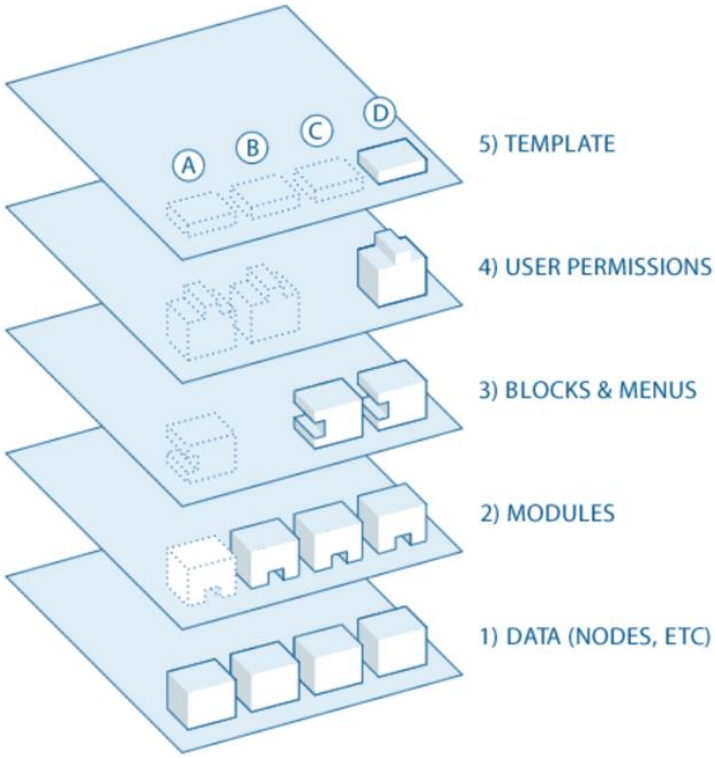

Figure 22. Drupal flow. [27.]

4.3 Drupal architecture

Drupal is a multifunctional Content Management System (CMS). It has ability to take care every aspect of the software development process to create a web application. Based on the architectural needs Drupal can be Monolithic or Traditional architecture and Decoupled architecture. [28.]

In Software Engineering, Monolithic architecture means a single software which has ability to take care all the aspect of its functionalities. A Monolithic architecture or Traditional architecture is single tiered where UI, code and data accessibility all works under one platform. In Monolithic approach to run a website, front-end and back-end are tightly coupled, therefore no need to rely on external facilities. [28.]

On the other hand, Decoupled architecture means separation of front-end from the back-end. In Decoupled Drupal architecture, User Interface (UI) is separated from the presentation part. The benefit of this architecture is that UI and Content Management will be provided by the Drupal and developers have the freedom to choose the presentation layer outside of Drupal. [28.]

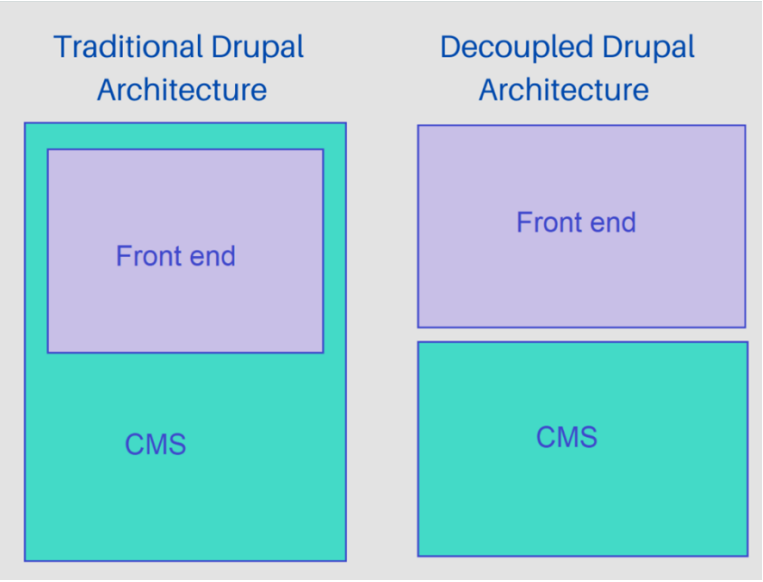

Figure 23. Traditional or Monolithic Drupal Architecture and Decoupled Drupal Architecture. [28.]

In a nutshell, in Drupal both Monolithic and Decoupled architecture follows the same structure when comes to back-end technology. It is the only presentation layer or front-end part which is the difference in these two architectures. [28.]

Different ways of leveraging Decoupled Drupal architecture: Initially, Drupal was developed based on Monolithic architecture. The benefit of this architecture is full control over the project. However, to leverage Decoupled architecture Drupal can be divided into two types. These are: Progressively Decoupled architecture and Fully Decoupled Architecture. [28.]

Progressively Decoupled Drupal architecture: The idea behind Progressively Decoupled architecture is that to have the opportunity to use of more JavaScript. The advantage of this approach is possibility of using both of Drupal's front-end action and JavaScript framework. However, the disadvantage of this approach is that it reduces Drupal's control over administration. [28.]

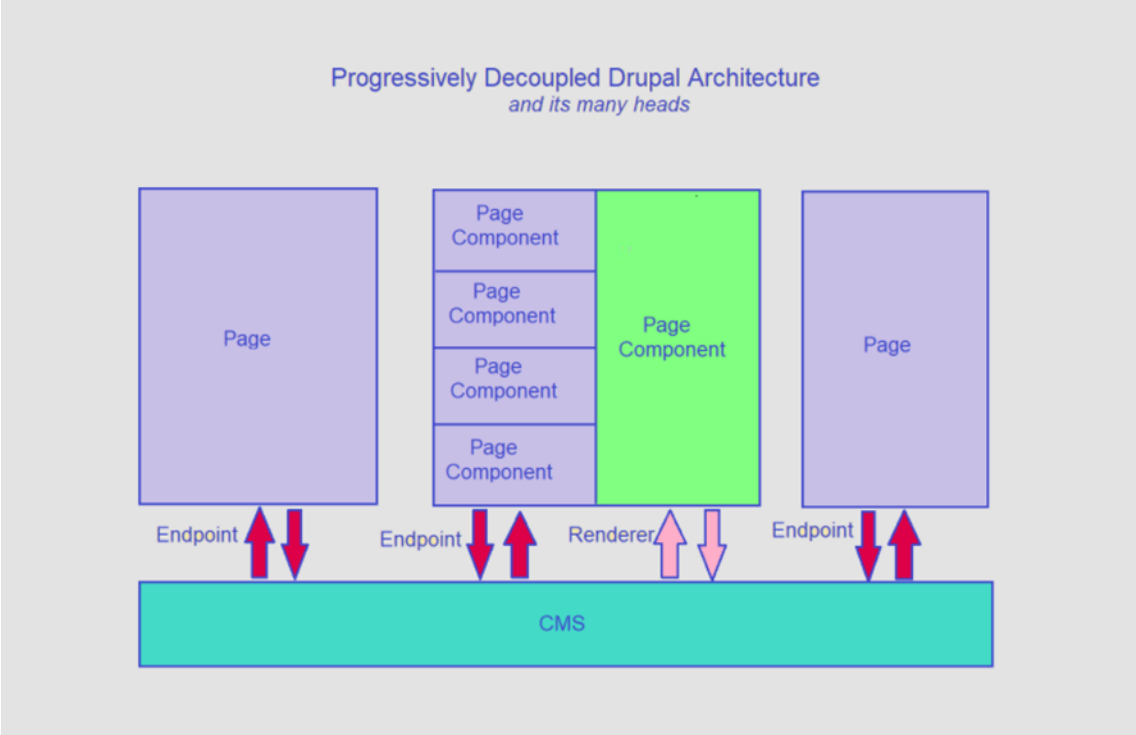

Figure 24. Progressively Decoupled Drupal Architecture. [28.]

Fully Decoupled Drupal Architecture: In this architecture there is full separation between client side and server side of Drupal application. The idea behind is separation of concern, as a result, client framework becoming a server-side pre-render and CMS becoming the data provider. [28.]

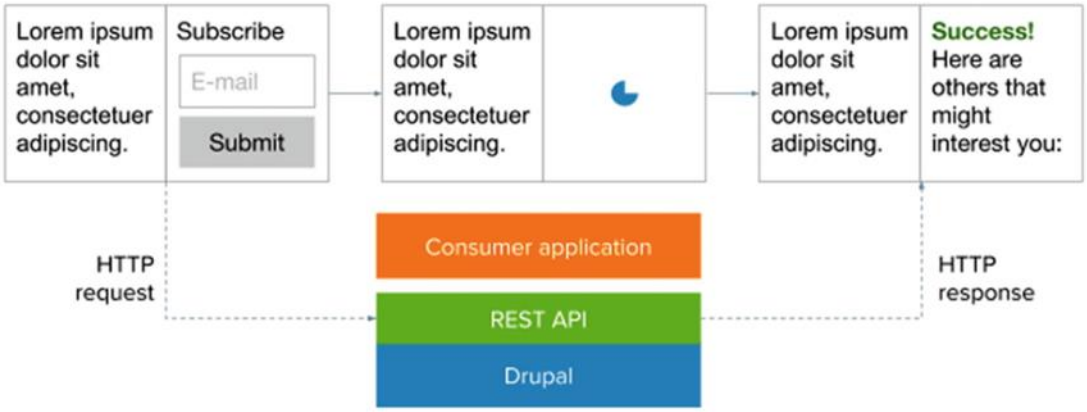

Figure 25. Fully Decoupled Drupal Architecture. [29.]

Here API works as a bridge between these two to retrieve data and provide it to the front end. The benefit of this approach is that Developers get additional control over the presentation layer by choosing their favorite front-end technologies. On the other hand, the disadvantage of this approach is many features such as ‘in-place editing’ will be unavailable. [28.]

5 Comparison between WordPress and Drupal

WordPress and Drupal are among the most widely used Content Management System (CMS) platform for website development. In this chapter author is going to discuss the comparisons between WordPress and Drupal in the areas of market share, security, installation, customization, ease of use, content management and performance.

5.1 Market share

WordPress is the most popular content management system. According to W3Techs, 40% of the websites are supported by WordPress on the internet, whereas 1.7% websites are powered by Drupal. In case of Content Management System's WordPress dominates the market even more. For example, 64% of CMS market share occupied by WordPress whereas only 3.0% is covered by Drupal. [30.]

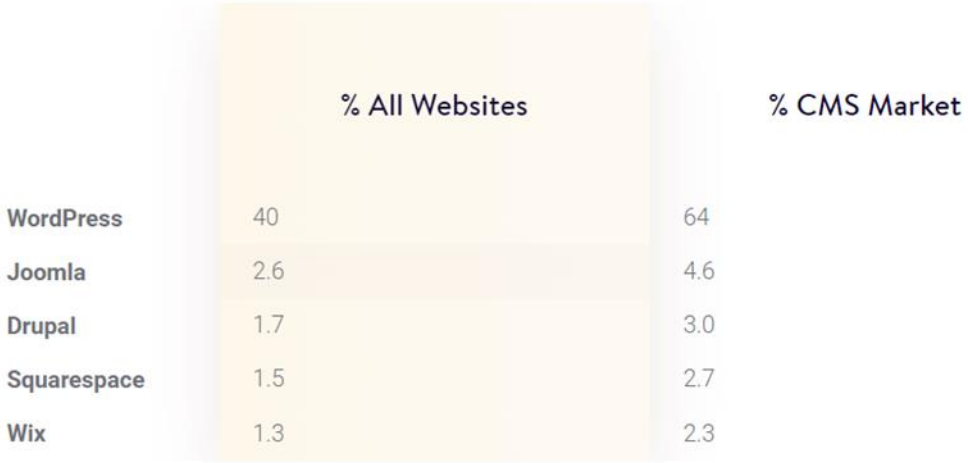

Figure 26. CMS market share. [30.]

5.2 Security

Both WordPress and Drupal are secure CMS. Based on website-built, customization and extension policy Drupal is ahead compared to WordPress.

Drupal has enterprise level security and site scale capacity. According to Sucri, in 2018, 90% of all hacked CMS are built in WordPress. The main security issue for WordPress is 3rd party plugins. When a plugin is upload to WordPress, due to non-verified 3rd party issues it become vulnerable. On the other hand, Drupal modules and themes are more secure for its internal security policy. [31.]

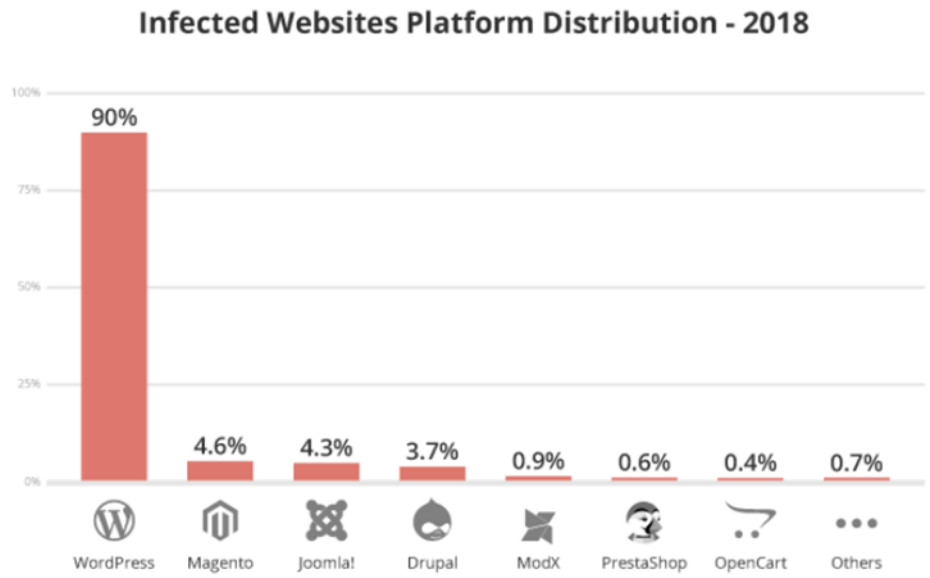

Figure 21. Infected Websites Platform Distribution. [31.]

5.3 Installation

WordPress is the quickest Content Management System to install. It takes only couple of minutes to install. WordPress does not require higher level of understanding about Web development. On the contrary, Drupal is little bit complex. It requires more steps compared to WordPress and higher level of understanding for installation. [32.]

5.4 Customization

WordPress offers a huge number of plugins and themes that can be useful to extend its features and functionalities. WordPress can create not only Blog Post but also News site, Portfolio site, eCommerce site et cetera. Drupal has also modules and extensions to extend its functionalities. However, due to its lesser number of extensions Drupal is not as customizable as WordPress. [33.]

5.5 Ease of use

WordPress is beginner friendly and easy to use. It does not require hard programming skills although it is useful to have programming skills. A wide range of plugins and themes makes a newbie’s life easier to develop a website. On the contrary, Drupal requires steep learning curve. It is not suitable for beginners. Drupal’s CMS is complex by nature. It demands at least some level of skills in HTML, CSS, PHP in order to maintain the website. [34.]

5.6 Content Management and Performance

WordPress and Drupal both can perform a lot of tasks in terms of content management. In the area of performance, Drupal is behind compared to WordPress. For instance, updating CMS in Drupal is more time consuming and difficult than WordPress. [33.]

	WordPress	Drupal
Installation	5min installation	10min – there are special installation packages
Appearance and extensions	Tens of thousands of extensions	Good number of extensions, harder to install
Blogging	Perfect blogging platform	Blogging module; not as good as WP
Support	Community, pay for support	Community, Marketplace
Security	Frequently updated, security plugins	Frequently updated, security plugins
Multilingual options	Should use 3rd party services	Activate language modules

Figure 22. WordPress and Drupal comparison table. [35.]

WordPress and Drupal both are great content management systems, and both have their own strengths and weakness. WordPress is ideal for basic website, easy customization, simple learning curve and plenty of themes and plugins. On the contrary, Drupal is suitable for complex website and advanced security.

6 Conclusion

Content Management System is a great platform for developing web contents which is relatively easy to maintain. It offers non-technical persons to publish their web page without knowing the standards of web programming. The main function of CMS is storing the content at the backend and at the same time publish the content through frontend interface. In this way individuals can save huge amount of investment and can focus on their core activities. As a result, it is becoming popular among the users who need quick solutions in terms of saving money and time for their online presence. The future of Content Management System relies on how it can cope with other technologies such as Chatbot, User Experience, Virtual and Augmented Reality.

References

1. Rouse, Margaret. Build a WCM architecture that supports business needs [online].
URL:https://searchcontentmanagement.techtarget.com/definition/content-management-system-CMS. Accessed 25 November 2019.

2. What is a Content Management System (CMS)[online]? 22 October 2019.
URL:https://kinsta.com/knowledgebase/content-management-system/.
Accessed 25 November 2019.

3. Barker, Deane. Web Content Management: The Component of Content Management Systems [online]. O'Reilly Media, Inc., 2016.
URL:https://learning.oreilly.com/library/view/web-content-managment/9781491908112/ch01.html.Accessed 25 November 2019.

4. Boiko, B. Content Management Bible, 2nd Edition. Indianapolis: Wiley; 2005.

5. Eden, B. L. Content Management Systems. Bingley: Bradford Emerald Group; 2006.

6. Heslop, Brent. History of Content Management Systems and Rise of Headless CMS [online]. 18 December 2018.
URL:https://www.contentstack.com/blog/all-about-headless/content-management- systems-history-and-headless-cms. Accessed 25 November 2019.

7. Ottervig, Vegard. The history of CMS – what has happened[online]? 3 June 2019.
URL: https://enonic.com/blog/the-history-of-cms--what-has-happened.
Accessed 25 November 2019.

8. Jones, Sydney. 5 Types of Content Management System (CMS)[online].
URL:https://www.ixiasoft.com/types-of-content-management-systems/.
Accessed 25 November 2019.

9. Winkels, Niklas. The Definitive Guide to CMS Architecture [online]. 29 July 2019.
URL: https://developers.bloomreach.com/blog/2019/cms-architecture.html.
Accessed 25 November 2019.

10. Muldoon, Kevin. What Is WordPress? What Can It Do & Is It Right For You? A Beginner's Guide [online]. 24 June 2021.
URL: https://www.wpkube.com/what-is-wordpress/. Accessed 8 November 2021.

11. The History of WordPress, its Ecosystem and Community[online]. 20 May 2021.
URL: https://kinsta.com/learn/wordpress-history/. Accessed 8 November 2021.

12. The History of WordPress from 2003 – 2021(with Screenshots) [online]. 19 June 2021.
URL:https://www.wpbeginner.com/news/the-history-of-wordpress/. Accessed 8 November 2021.

13. MacDonald, Matthew. WordPress: The Missing Manual, 3rd Edition. Sebastopol: O'Reilly Media, Inc; 2020.

14. Sonawane, Aashish. WordPress as a Headless CMS- What it is and why do you need it [online]?
URL:https://www.clariontech.com/blog/wordpress-as-a-headless-cms-what-it-is-and-why-do-you-need-it .Accessed 8 November 2021.

15. Sharma, Himanshu. WordPress Architecture Explained in Great Detail[online].
URL: https://www.optimizesmart.com/wordpress-ninja-15-minutes/. Accessed 8 November 2021.

16. Template files [online].
URL: https://developer.wordpress.org/themes/basics/template-files/. Accessed 8 November 2021.

17. Drupal Overview[online].
URL: https://www.softaculous.com/apps/cms/Drupal. Accessed 8 November 2021.

18. Drupal[online].
URL:https://whatis.techtarget.com/definition/Drupal. Accessed 8 November 2021.

19. Drupal[online].
URL:https://civicrm.org/drupal. Accessed 8 November 2021.

20. Drupal[online].
URL:https://drupal.com/ . Accessed 8 November 2021.

21. Our history[online].
URL:https://www.drupal.org/about/history. Accessed 8 November 2021.

22. Matic. History of Drupal[online]. 06 October 2016.

URL:https://www.agiledrop.com/blog/history-drupal. Accessed 8 November 2021.

23. From Drupal 1 to Drupal 8: an exciting journey to history[online]. 23 August 2017.
URL:https://drudesk.com/blog/evolution-from-drupal1-to-drupal8. Accessed 8 November 2021.

24. Matthews, Daniel. A brief history of Drupal from 1.0 to 8.0 [online]. 22 September 2016.
URL:https://opensource.com/life/16/9/drupal-evolution. Accessed 8 November 2021.

25. The History of Drupal Content Management System [online]. 13 April 2021.
URL:https://www.gwsmedia.com/articles/history-drupal-content-management-system. Accessed 8 November 2021.

26. VanDyk, John K. Pro Drupal Development, 2nd Edition. Berkely: Apress; 2008.

27. Overview [online]. 28 June 2021.
URL:https://www.drupal.org/docs/7/understanding-drupal/overview. Accessed 8 November 2021.

28. Kaur, Gurpreet. Monolithic and Decoupled Drupal Architecture [online]. 15 October 2020.
URL:https://opensenselabs.com/blog/articles/monolithic-and-decoupled-drupal-architectures. Accessed 8 November 2021.

29. So, Preston. Decoupled Drupal in Practice: Architect and Implement Decoupled Drupal Architectures Across the Stack. Ridgewood: Apress; 2018.

30. WordPress market share [online].
URL:https://kinsta.com/wordpress-market-share/. Accessed 8 November 2021.

31. Bassi, Ben. WordPress vs. Drupal: Which is Better [online]? 02 February 2021.
URL:https://www.commonplaces.com/blog/wordpress-drupal. Accessed 8 November 2021.

32. The main differences between WordPress and Drupal [online]. 14 July 2020.

URL:https://www.8p-design.com/en/blog/main-differences-between-wordpress-and-drupal. Accessed 8 November 2021.

33. Uzayr, Sufyan, Bin. WordPress vs Drupal: SEO, Security, Extensions, Content Management, All Compared [online]. 12 October 2020. URL:https://themeisle.com/blog/wordpress-vs-drupal-compared/. Accessed 8 November 2021.

34. Drupal vs. WordPress [online]. 21 December 2020. URL:https://www.ibm.com/cloud/learn/drupal-versus-wordpress-comparison. Accessed 8 November 2021.

35. Minaev, Artem. Drupal vs. WordPress Comparison 2021 [online].3 November 2021. URL:https://firstsiteguide.com/wordpress-drupal/. Accessed 8 November 2021.