



**Университет по библиотекознание и информационни
технологии**

Катедра: Информационни системи и технологии

Специалност: „Информационни технологии в медийния бизнес“

Магистърска теза

на тема:

Сравнителен анализ на трите най-популярни CMS-и - WordPress,
Drupal и Joomla

Дипломант:

Александър Иванов

Фак. №: 074-итмз

Научен ръководител:

Проф. д-р Иван Иванов

София

2018

ПРИЛОЖЕНИЕ №3

Декларация на дипломанта



УНИВЕРСИТЕТ ПО БИБЛИОТЕКОЗНАНИЕ И ИНФОРМАЦИОННИ ТЕХНОЛОГИИ

ДЕКЛАРАЦИЯ

От Александър Петров Иванов

(име, презиме и фамилия на студента)

Декларирам, че представената дипломна работа/магистърска теза е подготвена и изпълнена самостоятелно от мен.

При откриване на плагиатство поемам съответната отговорност по смисъла на чл. 31 (1-3) от Наредбата.

Дата: 07.03.2018г.

Подпис.....

(дипломант)

ПРИЛОЖЕНИЕ №. 7

Декларация за публикуване



**УНИВЕРСИТЕТ ПО БИБЛИОТЕКОЗНАНИЕ И ИНФОРМАЦИОННИ
ТЕХНОЛОГИИ**

ДЕКЛАРАЦИЯ

От Александър Петров Иванов

(име, презиме и фамилия на студента)

С настоящата отстъпвам / не отстъпвам безвъзмездно право на УНИБИТ да публикува представената моя авторска разработка (курсова работа, дипломна работа, магистърска теза) като общодостъпен ресурс за безплатен публичен достъп чрез информационните системи на УНИБИТ.

Дата: 07.03.2018г.

Подпис.....

(дипломант)

Съгласен съм авторската разработка (курсова работа, дипломна работа, магистърска теза) да бъде публикувана като общодостъпен ресурс за безплатен публичен достъп чрез информационните системи на УНИБИТ.

Дата: 07.03.2018г.

Научен ръководител

.....
(подпис)

РЕЗЮМЕ

НА МАГИСТЪРСКА ТЕЗА НА ТЕМА

„ Сравнителен анализ на трите най-популярни CMS-и - WordPress, Drupal и Joomla“

Иванов, А. Сравнителен анализ на трите най-популярни CMS-и – WordPress, Drupal и Joomla!. Научен ръководител проф. д-р Иван Иванов. 2018г. Катедра "Информационни системи и технологии". УНИБИТ. 73 с. Брой източници - 46, фигури - 12, таблици – 1.

Целта на магистърската теза е да се направи сравнителен анализ върху трите най-популярни системи за управление на съдържание с помощта на теоретичен преглед върху материята и примерни уебсайтове показващи техните предимства и недостатъци. Във връзка с това се поставят следните *задачи*:

- Прави се характеристика на управлението на съдържанието като процес;
- Разглежда се същността и значението на системите за управление на съдържанието (CMS) и се предлагат критерии за техния избор;
- Анализират се системите за управление на съдържанието като цяло, тяхната архитектура, имплементиране и различните им видове;
- Описват се и се сравняват трите най-използвани CMS с отворен код – WordPress, Drupal и Joomla, чрез представяне на уебсайтове, изградени върху тези платформи.

В *първа глава* на разработката се анализират теоретичните основи на управлението на съдържанието като процес и същностната характеристика на системите за управление на съдържанието. Разглеждат се значението, архитектурата, имплементацията на CMS, както и работния процес на управление на съдържанието.

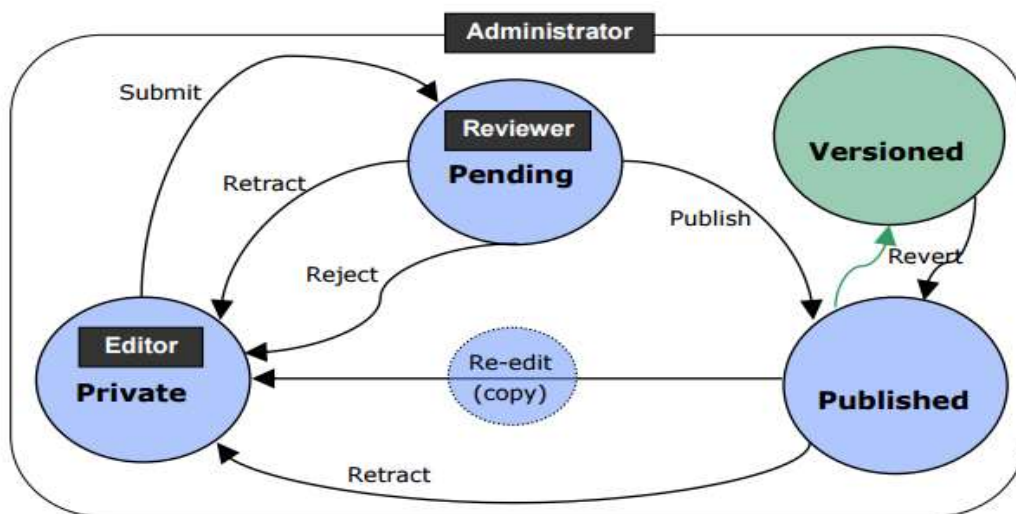
Втора глава на тезата е разделена на две основни части. В първата се прави подробно описание на трите най-разпространени CMS, – Joomla!, WordPress и

Digital, а във втората те се сравняват по различни показатели и с помощта на примерни уебсайтове накрая резултатът се обобщава.

Управлението на съдържанието е съвкупност от процеси и технологии, която поддържа събирането, управлението и публикуването на информация във всякаква форма или среда. Когато се съхранява и използва чрез компютри, тази информация може да бъде по-конкретно определена като цифрово съдържание, или просто - съдържание. Дигиталното съдържание може да бъде под формата на текст (например електронни документи), мултимедийни файлове (например аудио или видео файлове) или всеки друг тип файл, който следва работния процес, изискващ управление.

Управлението на съдържанието е съвместен процес. То често включва ролите и отговорностите показани на фигурата по-долу:

- Автор на съдържание;
- Администратор;
- Редактор;
- Издател.



Работен процес на управлението на съдържанието

Терминът “система за управление на съдържанието” първоначално е бил използван за системи за публикуване и управление на уебсайтове. Системата за управление на съдържанието е компютърен софтуер, който подпомага своите

потребители в процеса на управление на съдържанието. Тя улеснява организацията, контрола и публикуването на големи количества документи и друго съдържание като изображения и мултимедийни ресурси.

Основният критерий, по който CMS могат да се разделят е според степента на откритост и софтуерния лиценз. В изложението на магистърската теза са разгледани подробно предимствата и недостатъците на системите с отворен, затворен код, както и тези с платен и свободен лиценз.

В таблицата на следващата страница е представена в синтезиран вид сравнителна характеристика на трите най-популярни CMS с отворен код - Joomla!, Drupal, WordPress.

Сравнение между основните характеристики на Joomla!, Drupal, WordPress

Характеристики	Drupal	Joomla!	WordPress
Сложност при използване	Средна	Средна	Малка
Оптимизация на търсенето	ДА, изисква се добавка	ДА, изисква се плъгин	ДА
Благоприятна среда за разработчиците	ДА	ДА, по-ограничена от Drupal	НЕ
Управление на множество сайтове с една инсталация	ДА	НЕ	НЕ, изисква WordPress MU
Блогове	ДА	ДА, по-ограничена от WordPress	Да, отлична платформа за блогове
Отделна област за администриране	ДА	ДА	ДА
Поддръжка от общността	ДА	ДА	ДА
Мащабируемост	ДА	ДА	НЕ, обикновено се използва за сайтове с еднородно съдържание

WordPress се намира в единия край на диапазона като най-лесен за употреба, но неподходящ за комплексни проекти, Joomla! е в средата, и Drupal е на другия

край на диапазона като най-труден за използване и най-подходящ за сложни проекти.

На база на изложението, представено в магистърската теза могат да се направят следните *основни изводи*:

- Управлението на съдържанието е основна част от всяка организация, като един CMS продукт може да покрие множество категории на съдържанието;
- Drupal е подходяща за информативни и интранет уебсайтове, тъй като изисква по-малко ресурси; Drupal предполага по-голяма гъвкавост, но администрацията е доста сложна за уебмастъра. Клиентите трябва да наемат експерт в Drupal и общите потребители на сайта не могат лесно да бъдат обучени да използват Drupal, както могат да бъдат обучени, за да използват Joomla!.
- За интранет сайтове, с множество обекти се препоръчва използването на Joomla!, понеже тя се справя с натоварването по-добре от Drupal, особено след кеширане; Joomla има много по-голяма общност от разработчици в сравнение с Drupal.
- WordPress се оказва най-универсалната CMS от трите разглеждани в магистърската теза, най-вече при сайтове, които са създадени основно с цел да показват информация, такива, които трябва да бъдат гъвкави, за да отговарят на променящите се в реално време нужди на потребителите или сайтове за електронна търговия.

С Ъ Д Ъ Р Ж А Н И Е

Увод	8
ГЛАВА I. АНАЛИЗ НА ПРОЦЕСА И СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА СЪДЪРЖАНИЕТО	12
1. Управление на съдържанието	12
1.1. Работен процес на управлението на съдържанието	12
1.2 Реализация и експлоатация на процеса на управление на съдържанието	14
2. Системи за управление на съдържанието – същностна характеристика, архитектура и връзка с останалите управленски системи в организацията	15
2.1 Основни характеристики и възможности на системите за управление на съдържанието	17
2.2 Архитектура на системите за управление на съдържанието	19
2.3 Връзка между действащите системи, изградени в организацията	20
3. Избор и имплементиране на системите за управление на съдържанието	21
3.1 Бизнес цели и стратегии	21
3.2 Изисквания към системите за управление на съдържанието	22
4. Основни видове системи за управление на съдържанието – предимства и недостатъци	27
ГЛАВА II. ХАРАКТЕРИСТИКА И СРАВНИТЕЛЕН АНАЛИЗ НА СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА СЪДЪРЖАНИЕ С ОТВОРЕН КОД – DRUPAL, WORDPRESS, JOOMLA!	35
1. Преглед на Drupal, WordPress, Joomla!	35
1.1 Drupal	35
1.2 WordPress	41
1.3 Joomla!	44

2.Сравнение на системите за управление на съдържанието с отворен код – Drupal, WordPress и Joomla!	49
2.1 Примерни уебсайтове изградени върху Drupal, WordPress и Joomla	61
Заклучение	69
Използвана литература	71

Увод

Напредъкът в областта на научноизследователската и развойна дейност доведоха до развитието на технологии като: управление на съдържанието, разпределени изчисления, извличане на данни и обработка. Въпреки че терминът „управление на съдържанието“ е сравнително нов, практиките свързани с него датират от края на 80-те години на ХХ в. Времевата рамка между края на 1995 г. и 1996 г. е тази, с най-агресивните периоди на растеж на Интернет. Огромният потенциал на пазара е бил очевиден за всички в сектора на информационните технологии. Фирмите се стремели с всички сили да изградят уебсайтове, за да се конкурират на възникващия пазар на Интернет, което означавало, че те трябвало да впрегнат техните съществуващи ресурси и да ги комбинират с новите интернет технологии. В резултат на това, необходимостта от управление на съдържанието дошло на преден план.

Управлението на съдържанието може да бъде от голяма полза за бизнеса, особено електронния. Този процес подобрява технологиите на електронния бизнес, докато в същото време играе ключова роля като част от решението за управлението на данните. Поради тази причина, поддържането и използването на квалифицирана и гарантирана информация подлежи на по-специално управление и координация поради високото си комерсиално значение. Основни потребители на такъв род информация се явяват бизнес потребителите. Те са тези, които се нуждаят от сигурност и надеждност на данните. Системите за управление на съдържанието

(CMS) спомагат за разрешаването на подобни проблеми, като автоматизират и улесняват процеса на добавяне и промяна на съдържанието.

Системите за управление на съдържанието помагат да се създаде и съхрани съдържание в споделено хранилище. Освен това, те управляват автономно взаимоотношенията между елементите на съдържанието (например следене на това, къде те се вписват в йерархията). Някои CMS също така предоставят способности за проследяване на състоянието на елементите на съдържанието чрез редакционни процеси и работни потоци.

С помощта на CMS, редакторите на съдържание и организациите, за които те работят, получават редица предимства.

Например, това може да са:

- Координиране на работата на екипи от автори и редактори (например, като се гарантира, че само един човек редактира всеки отделен елемент от съдържанието във всеки един момент).
- Контролиране на брандирането и качеството на съдържанието (например, като се гарантира, че промените в съдържанието са одобрени преди те да бъдат публикувани).
- Използване на един и същи елемент на съдържанието в множество различни формати.

Във втората половина на уводната част ще се посочи накратко съдържанието в двете глави на магистърската теза, целта и задачите, подходите към изследването и ограниченията, които се поставят.

В първа глава на магистърската теза се анализират теоретичните основи на управлението на съдържанието като процес и същностната характеристика на системите за управление на съдържанието. Разглеждат се значението, архитектурата, имплементацията на CMS, както и работния процес на управление на съдържанието.

Втора глава на разработката е разделена на две основни части. В първата се прави подробно описание на трите най-разпространени CMS – Joomla!, WordPress и

Drupal, а във втората те се сравняват по различни показатели, като накрая резултатът се обобщава.

Цел и задачи на магистърската теза

Целта на магистърската работа е да се направи сравнителен анализ на трите най-популярни системи за съдържание с помощта на теоретичен преглед върху материята и примерни уебсайтове. Във връзка с това се поставят следните *задачи*:

- Прави се характеристика на управлението на съдържанието като процес;
- Разглежда се същността и значението на CMS и се предлагат критерии за техния избор;
- Анализират се системите за управление на съдържанието като цяло, тяхната архитектура, имплементиране и различните им видове;
- Описват се и се сравняват трите най-използвани CMS с отворен код – WordPress, Joomla! и Drupal.

Подходи към изследването

За постигане целта на изследването се използват ресурсите на интердисциплинарния, системния, комплексния и сравнителния подход.

Интердисциплинарният подход намира израз в изследването на процесите и явленията от позициите на потребителите на системите за управление на съдържанието и от позициите на разработчиците на тези системи.

Системният подход намира израз в разглеждането на управлението на съдържанието като система в отчитането на връзките и зависимостите между степента на удовлетвореност на крайния потребител и вложените усилия от страната на разработчиците, между средствата и ресурсите, необходими за разработването и поддържането на системи за управление на съдържанието и крайния резултат и т.н.

Комплексният подход намира израз в изследването на всички елементи на процеса на управление на съдържанието и CMS.

Сравнителният подход се използва при анализирането на предимствата и недостатъците на различните видове CMS и следователно улесняване на преценката коя CMS ще бъде подходяща в даден случай.

Ограничения

При разработването на магистърската теза се поставят следните ограничения:

- Анализът в магистърската теза набляга върху уеб системите за управление на съдържанието като най-популярни. Затова под CMS ще се разбира именно това подразделение на системите за управление на съдържанието. Останалите видове се споменават накратко в Глава I.
- Въпреки че в теоретичната част се разглеждат различни видове системи за управление на съдържанието, примерите, разгледани в Глава II на магистърската теза са част от CMS с отворен код, тъй като те са най-широко използвани.

ГЛАВА I. АНАЛИЗ НА ПРОЦЕСА И СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА СЪДЪРЖАНИЕТО

1. Управление на съдържанието

Управлението на съдържанието е съвкупност от процеси и технологии, която поддържа събирането, управлението и публикуването на информация във всякаква форма или среда.¹ Когато се съхранява и използва чрез компютри, тази информация може да бъде по-конкретно определена като цифрово съдържание, или просто - съдържание. Дигиталното съдържание може да бъде под формата на текст (например електронни документи), мултимедийни файлове (например аудио или видео файлове) или всеки друг тип файл, който следва работния процес, изискващ управление.² В следващите редове ще разгледаме по-подробно този процес.

1.1. Работен процес на управлението на съдържанието

Управлението на съдържанието е съвместен процес. То често включва следните основни роли и отговорности (Фиг.1):

Автор на съдържание - това е специална роля, давана на потребителите, когато те създават даден елемент (например уеб страница, папка и т.н.). Тя се отнася само за един елемент, като информацията се съхранява върху него. Обикновено системата възлага тази роля автоматично.

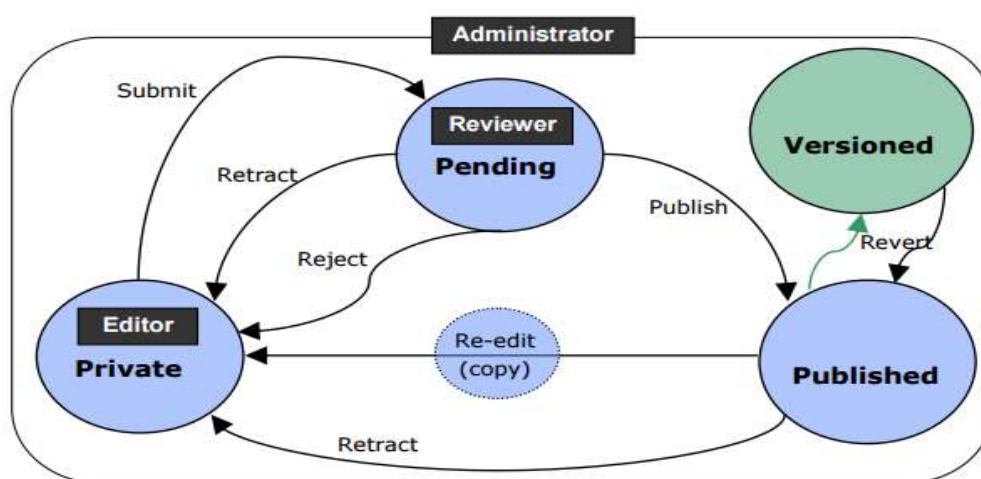
Администратор - тази роля се дава на тези потребители, които ще управляват сайта. Отговорен е за запазването на съдържанието в хранилище, така, че по-късно то да може да бъде намерено и използвано. Администраторът също управлява потребителите и правата, които има всеки от тях върху съдържанието – той може да изтрива или редактира съдържание, добавя / премахва потребителите и до известна степен да променя конфигурацията на даден уеб сайт.

¹ Boiko, B., Understanding Content Management, Metatorial Services Inc. & HungryMinds Inc, 2012, p.43, 07.03.2018 http://www.metatorial.com/downloads/Boiko_Wp_UnderstandingContentManagement.pdf

² Pico, P., Silva, A., WORKFLOWS IN CONTENT MANAGEMENT SYSTEMS, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 2008, pp.122-126, 07.03.2018 <http://isg.inesc-id.pt/alb/static/papers/2008/i71-pdsp-webist2008.pdf>

Редактор - отговорен за проверката на съдържанието и за начина му на поднасяне на читателите. Редакторите не могат да публикуват в уеб сайта в реално време. Те представят техните промени до издателя, който или ги одобрява за публикуване или ги отхвърля и ги изпраща за поправка.

Издател - това е потребител с ограничени разрешения, чиято роля е да направи преглед на съдържанието, представено от редактора и да реши, кога то е готово за публикуване. Работният процес добавя елемент за управление на качеството, както и „предпазна мрежа“, която гарантира, че невъзможността да се редактира уеб страницата в реално време. Все пак, трябва да се има предвид, че последните две роли могат да бъдат комбинирани, което дава възможност на техните притежатели да създават съдържание и да го публикуват веднага.



Фиг.1 Работен процес на управлението на съдържанието³

Работният процес е процесът на създаване на цикли от последователни и паралелни задачи, които трябва да бъдат изпълнявани в системата за управление на съдържанието. Системите за управление на съдържанието предлагат средства за управление на жизнения цикъл на документа от времето на първоначалното му създаване до неговата проверка, публикуване, архивиране и унищожаване. Например авторът на съдържанието добавя статия, но тя не се публикува на сайта

³ Managing and maintaining a CMS website, University of Bristol, Reference Guide, 2010, p. 8, 07.03.2018 <http://www.bristol.ac.uk/is/media/training/documentation/webcms-1/webcms-1r.pdf>

докато редакторът не я провери и главният редактор не я одобри. Това често е свързано с обмен на съобщения

За да обобщим, работният процес е последователност от действия или събития прилагани към даден обект (уеб страница, изображение, папка, и т.н.), с цел да се трансформира, докато достигне желаното крайно състояние. Фигурата по-горе показва различните роли, които могат да бъдат причислени към даден потребител, различните състояния на даден елемент и взаимоотношенията между тях.

1.2 Реализация и експлоатация на процеса на управление на съдържанието

Реализациите на управлението на съдържанието трябва да са в състояние да управляват разпространението на съдържание и дигиталните права в жизнения цикъл на съдържанието.⁴ Системите за управление на съдържанието обикновено участват с управлението на дигиталните права, за да контролират достъпа на потребители до тях. В този случай, структурите на системите за управление налагат някои ограничения върху управлението на съдържанието, тъй като те не позволяват на авторите да го променят по време на неговия жизнен цикъл. Създаване на ново съдържание с помощта на управлявано (защитено) съдържание също е проблем, поради който защитеното съдържание остава извън системите за контролирано управление. Малко са реализациите на управлението на съдържанието, които решават всички тези проблеми.

Организациите се нуждаят от управление на съдържанието, ако:

- използват едно и също съдържание многократно в една публикация (например предупреждение в указанията за безопасност могат да се използват много пъти в самото упътване);
- използват еднакво съдържание в повече от една публикация или документ (например номер на част и нейното описание в каталог за части могат да се използват в ръководството на потребителя);

⁴ Halverson, K., Rach, M., Content Strategy, New Riders, 2nd edition, 2012, p.41, 07.03.2018
<http://ptgmedia.pearsoncmg.com/images/9780321808301/samplepages/0321808304.pdf>

- публикуват тяхното съдържание в повече от един медиен канал (например съдържанието в принтирано упътване за работа, което се доставя с продукта, може да бъде използвано и в онлайн помощната информация);
- публикуват съдържанието си на много езици.

Видовете публикации, които тези организации съставят могат да включват:

- техническа информация (каталози за части, документация за софтуер, потребителски ръководства);
- референтни материали (енциклопедии, речници, стандартни ръководства);
- изпитни и учебни материали (програми за електронно обучение, изпитни материали);
- образователни и маркетингови материали (опаковане, промоционални рекламни листовки, брошури).

2. Системи за управление на съдържанието – същностна характеристика, архитектура и връзка с останалите управленски системи в организацията

Системата за управление на съдържанието е компютърен софтуер, който подпомага своите потребители в процеса на управление на съдържанието. Тя улеснява организацията, контрола и публикуването на големи количества документи и друго съдържание като изображения и мултимедийни ресурси.⁵ Системата за управление на съдържанието улеснява съвместното създаване на документи от различни потребители.

Терминът „система за управление на съдържанието” първоначално е бил използван за системи за публикуване и управление на уебсайтове. Ранните системи за управление на съдържанието са били разработвани вътрешно в организации,

⁵ Mican, D., Tomai, N., Coroş, R., Web Content Management Systems, a Collaborative Environment in the Information Society, Informatica Economică vol.13, no 2/2009, p. 78, 07.03.2018 <http://revistaie.ase.ro/content/50/003%20-%20mican.pdf>

които са публикували голямо количество материали в уеб пространството, като онлайн магазини, вестници и корпоративни бюлетини.

Системата за управление на съдържанието се различава от софтуера за изграждане на уебсайтове като Microsoft FrontPage или Macromedia Dreamweaver по това, че позволява на потребители с малко или никакъв опит да правят промени в съществуващ уебсайт. Инструментите за изграждане на уебсайтове като FrontPage и Dreamweaver изискват повече технически познания и опит. Системата за управление на съдържанието е лесен за използване инструмент, който дава на упълномощените за това потребители възможността да управляват уебсайта.⁶ Тази система е инструмент за поддържане на уебсайт, а не за създаването му. При тези системи се използва стандартът CMIS (Content Management Interoperability Services). Други стандарти са JSR 170 (Java Content Repository) и JSR 283.

Основната функция и използването на системи за управление на съдържанието е да се представи информация в уебсайтове. Има изключително голям избор от такива системи и всяка една от тях е различна сама по себе си. Едни от тях предоставят само необходимите за обикновения потребител функции по лесен и достъпен начин, а други - възможност за пълен контрол (обработка) на данните. Повечето системи за управление включват публикуване, форматиране, преглед, индексирание, търсене и извличане на текстове. Системата може да се използва за централно хранилище, съдържащо документи, филми, снимки, телефонни номера, научни данни и др. Тя може да се използва за съхранение, контролиране, преразглеждане, семантично обогатяване и публикуване на документацията.⁷

⁶ Michelinakis, D., Open Source Content Management Systems: An Argumentative Approach, The University of Warwick Warwick Manufacturing Group, 2004, pp.209-211, 11.01.2018 <http://www.henworx.de/pdf/oscms-report.pdf>

⁷ Пак там.

2.1 Основни характеристики и възможности на системите за управление на съдържанието

Въпреки различните видове системи за управление на съдържанието (ще бъдат разгледани в отделна част от настоящата глава), те имат общи характеристики. Основните такива са:

Автоматизирани шаблони. Създаване на стандартни шаблони за извеждане, които могат да се прилагат автоматично за ново и съществуващо съдържание, което позволява да се промени облика на цялото съдържание от едно централно място.

Контрол на достъпа. Някои системи за управление на уеб съдържание поддържат потребителски групи. Потребителските групи позволяват да се контролира взаимодействието на регистрираните потребители със сайта. Дадена страница на сайта може да бъде ограничена до една или повече групи. Това означава, че за определени потребители може да бъде отказан достъп до страницата.

Разширяемост. Повечето съвременни CMS имат възможност за разширяване в зависимост от настройките на сървъра. CMS може създават микросайтове / уеб портали в рамките на основния сайт.

Лесна редактируемост на съдържанието. След като съдържанието е отделено от визуалното представяне на даден сайт, то обикновено може много по-лесно и по-бързо да се редактира и манипулира. Повечето CMS включват инструменти за редактиране, позволяващи на нетехнически потребители да създават и редактират съдържанието.

Мащабируеми функции. Повечето CMS включват плъгини или модули, които могат лесно да се инсталират, за да разшири функционалността на съществуващ сайт.

Обновяване на уеб стандартите. Активните CMS обикновено получават редовни актуализации, които включват нови възможности и поддържат системата до актуалните уеб стандарти.

Управление на работния процес. Създаване на цикъл на последователни и паралелни задачи, които трябва да бъдат изпълнени в CMS. Например, един

или много творци могат да представят една история, но тя не се публикува докато главния редактор не я одобри.

Сътрудничество. CMS софтуера може да действа като платформа за сътрудничество, позволяваща със съдържанието да работят един или много упълномощени потребители. Промените могат да бъдат проследени и разрешени за публикуване или игнорирани и върнати към стари версии.

Упълномощаване. Някои CMS дават възможност за различни групи потребители да разполагат с ограничени привилегии над специфично съдържание на сайта и носене на отговорност за управление на съдържанието.

Управление на документи. CMS софтуера може да осигури средства за съвместно управление на жизнения цикъл на документ от първоначалното създаване, преразглеждане, публикуване, архивиране и унищожаване на документ.

Виртуализация на съдържанието. CMS софтуера може да предостави средство, което позволява на всеки потребител да работи във виртуално копие на целия сайт, документ.

Разпространение на съдържание. CMS често съдейства за разпространение на съдържание чрез генериране на RSS и Atom потоци от данни към други системи.

Многоезичност. Възможност за показване на съдържание на множество езици.

Версии. CMS софтуера може да запазва различни версии на документите, което позволява на оторизираните редактори да извлекат предишните версии и да продължат работата от избрана точка, защото може да се наложи да се върне предишно копие.

Основните възможности на системите за управление на съдържанието (Web) са:

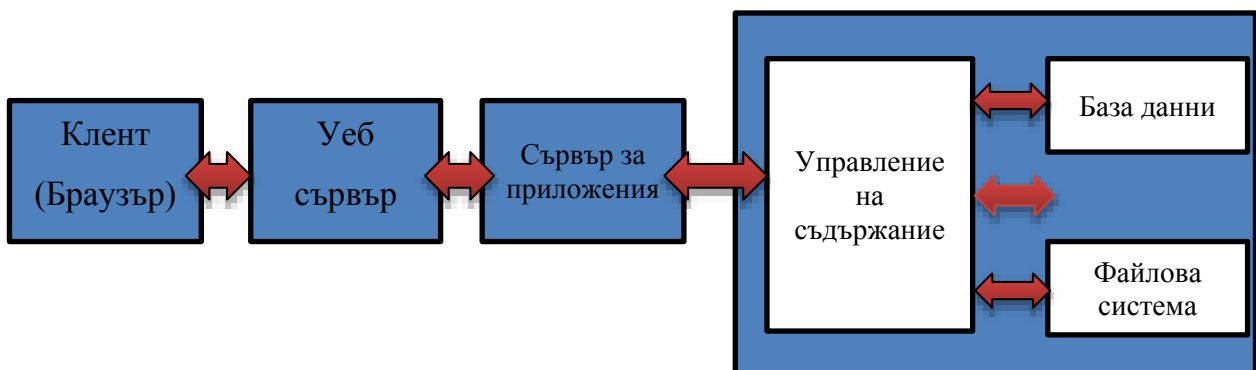
- Използване на шаблони за изобразяване, автоматично прилагани към нов или съществуващ контент;
- Простота на редактиране на контента, благодарение на визуален редактор. От потребителите не се изисква познаване на езици за програмиране;

- Възможност за разширяване на функционалността чрез установяване на допълнителни модули и плъгини (програми, които допълват функционалността на дадено основно приложение, добавяйки някакви нови възможности в областта на мултимедията);
- Визуализация на контента, благодарение на което той може да се преглежда и коригира преди инсталирането му на сайта.
- Управление на документооборота;
- Единна категоризация всички видове съдържание;
- Управление на потребители с различно нива на достъп;

2.2 Архитектура на системите за управление на съдържанието

За да се разбере по-добре как функционират системите за управление на съдържанието ще разгледаме тяхната архитектура (Фиг. 2).

В основата на CMS технологията стои архитектура от типа клиент/сървър на три нива – обработка на данни от клиента (уеб браузър), уеб сървър и сървър за приложения и съхранение на данни. За разлика от традиционната архитектура от две нива, тук присъства сървър за приложения като посредник между клиента и хранилището на данни. В системата има две хранилища. В първото хранилище (обикновено релационна система за управление на база от данни) се съхраняват всички данни, които са публикувани на интернет страницата. Във второто (обикновено файлова система) се съхраняват елементите – шаблони, графики и др.⁸



⁸ Luinenburg, L., Jansen, S., Designing Web Content Management Systems Using the Method Association Approach, Department of Information and Computing Sciences, Utrecht University, The Netherlands, 2008, p.16, 07.03.2018 <http://ceur-ws.org/Vol-389/paper08.pdf>

Фиг.2 Архитектура на системата за управление на съдържанието⁹

Архитектурата работи по следния начин:

- 1) Уеб браузърът изпраща HTTP заявка за определена уеб страница.
- 2) Уеб сървърът получава заявката, извлича файла и го подава на сървъра на приложения.
- 3) Сървърът за приложения започва да изпълнява програмния код във файла и се свързва с хранилищата на данни и му изпраща заявка.
- 4) Хранилищата получават заявката, обработват я и изпращат резултатите обратно на сървъра за приложения.
- 5) Сървърът за приложения довършва програмния код, което обикновено включва форматиране на резултатите в HTML. След това връща получения резултат на уеб сървъра.
- 6) Уеб сървърът изпраща на потребителя само резултата, във вид на HTML файл.

По този начин сървърът за приложения играе ролята на стандартизирана платформа за доставяне на динамично съдържание. Сървърите за приложения могат да бъдат много и връзката помежду им е чрез уеб сървър. Сървърните приложения са написани на скриптов езици или инструменти като Coldfusion, PHP, Perl, JSP, ASP, Python които обработват данните и ги превръщат във визуално съдържание.

2.3 Връзка между действащите системи, изградени в организацията

Съществува връзка между системите за управление на съдържанието и другите информационни системи в организацията, включително:

- Управление на документи;
- Управление на записи;
- Управление на електронна информация.

Тези системи обикновено се продават поотделно и постигането на съвместимост между тях не е лесно. Напредък в това отношение постига

⁹Фигурата е адаптирана от <http://www.sitecore.net/Learn/Blogs/Technical-Blogs/John-West-Sitecore-Blog/Posts/2013/10/Oversimplified-Conceptual-Architecture-of-the-Sitecore-ASPNET-CMS.aspx>

създаването на така наречените системи за управление на фирмено съдържание (Enterprise Content Management Systems – ECMS), които комбинират централна система за управление на съдържанието с други инструменти за управление на пълния набор от информация съществуваща в рамките на организацията.¹⁰ Тези системи са все още относително недоразвити и не е постигнато съгласие по въпроса какви възможности трябва да включват. В следващите няколко години се очаква рязко разширяване на пазара на такива системи и на покачване на интереса към тях.

В момента на пазара съществува огромен брой системи за управление на съдържанието, които имат различни възможности и силни страни. Затова всяка организация, която желае да внедри и използва CMS трябва да отдели време да определи своите бизнес правила и изисквания към системата и след това подробно да изследва и оцени предимствата и недостатъците на съществуващите в момента на пазара CMS и доколко те са подходящи за нейните цели и дейности. Отделните организации имат уникален набор от изисквания към CMS, затова не съществува решение, което да удовлетвори напълно всички от тях. По тази причина ръководителите трябва да отделят достатъчно време и средства за проучвания, за да изберат относително най-доброто решение от съществуващите на пазара.

3. Избор и имплементиране на системите за управление на съдържанието

В тази част от Глава I ще бъдат разгледани изискванията, на които трябва да отговаря една система за управление на съдържанието, за да бъде имплементирана в дадена организация.

3.1 Бизнес цели и стратегии

Основният въпрос, на който трябва да си отговори всяка организация, желаеща да използва система за управление на съдържанието е защо ще се използва CMS. Преди да се установят индивидуалните изисквания към CMS трябва да се определят бизнес целите, които ще бъдат постигнати чрез внедряването ѝ. Те трябва да отговарят на дългосрочните стратегии и насоки на бизнеса. Всяко

¹⁰ Enterprise Content Management (ECM) Overview, A White Paper, Intergraph, 07/2011, p.9, 12.02.2018
<http://www.intergraph.com/assets/plugins/ppmcollaterals/files/EMCOverview.pdf>

изискване трябва да е свързано с една или повече бизнес цели. Това позволява да се идентифицират причините за изискванията и да се подредят по приоритет. Добра идея е в зависимост от големината и структурата на организацията, в екипа, който ще оценява и ще прави избор на система за управление на съдържанието да се включат членове от различни отдели, тъй като създаването на съдържанието, използването и поддръжката на системата често се извършват от цял екип и за различни секции от сайта отговарят различни хора.

Най-важно е да се определи вида на съдържанието, което ще се публикува. Това влияе пряко върху функционалността, която ще се изисква от системата. Основният въпрос, на който трябва да си отговори всяка организация, желаеща да използва система за управление на съдържанието е защо ще се използва CMS.¹¹

Не съществува единствен най-добър списък с изисквания към CMS. Всяка организация има уникални потребности. Затова е добре да се използва структуриран подход за определяне на списъка с изисквания, за да се гарантира, че той ще бъде едновременно управляем и достатъчен.

3.2 Изисквания към системите за управление на съдържанието

Изискванията към системите за управление на съдържанието от страна на организацията, в която ще се имплементират могат да се групират в следните категории:¹²

- ❖ Създаване на съдържание;
- ❖ Управление на съдържание;
- ❖ Публикуване;
- ❖ Презентация;
- ❖ Бизнес изисквания.

Нека да разгледаме поотделно изискванията за всяка категория.

¹¹ How to evaluate a content management system, 12.02.2018 http://www.steptwo.com.au/papers/kmc_evaluate/

¹² Merker, L., A Guide to Web Content Management System Evaluation, White Paper, pp.3-7, 12.02.2018 https://omniupdate.com/resources/pdfs/white_papers/cms_evaluation_guide.pdf

1) Създаване на съдържание

Създаването на съдържание е функционалността, от която се нуждаят авторите на съдържание, използвайки CMS. Ключови изисквания от тази категория са:

Интегрирана среда за създаване на съдържание: CMS да осигурява мощна и удобна среда за авторите на съдържание. Тя им осигурява лесен достъп до пълния набор от възможности, осигурени от системата;

Разделяне на съдържанието от презентацията: Не е възможно да се публикува съдържание в множество формати без стриктно разделяне на съдържанието от презентацията. Форматирането на съдържание трябва да е базирано на стилове и да се извършва по време на публикуване, а не при неговото създаване;

Поддържане на множество потребители: CMS ще има много потребители, работещи по едно и също време. Затова тя трябва да осигурява възможности като потребителска автентикация и заключване на записите, за да се избегнат едновременни промени в съдържанието;

Single-sourcing (преизползване на съдържанието): Една и съща страница или дори параграф често са използвани в различен контекст или предоставяни на различни групи от потребители. Това е предпоставка за управление на различни платформи (Интернет, интранет) от един и същ източник на съдържание;

Създаване на метаданни: Използването на мета данни (автор, заглавие, дата на публикуване, ключови думи и др.) е много важно при управлението на големи хранилища от данни. То включва също индексирание на ключови думи, групиране и систематизиране на съдържанието;

Гъвкаво създаване на хипервръзки: Авторите създават множество хипервръзки между страниците и те трябва да остават стабилни при реструктуриране;

Нетехническо създаване на съдържание: Не трябва да се изисква от авторите да използват HTML (или други технически познания) докато създават страници;

Лесно използване и ефективност: За да бъде успешна една CMS създаването и управлението на съдържанието трябва да бъдат лесни и удобни.

2) Управление на съдържание

В основата на повечето CMS решения е централно хранилище, съпътствано от набор от инструменти за манипулиране и управление на съдържанието. Ключови изисквания от тази категория са:

Контрол на версиите и архивиране: Стриктният контрол на версиите е необходим за архивиране и възстановяване на съдържанието при срив на системата;

Работен процес: Децентрализираното създаване на съдържание разчита силно на мощен модел на работния процес, който да може да бъде лесно настроен и да е устойчив на организационни промени;

Сигурност: Да са осигурени адекватни нива на сигурност, проследяване и контрол за да се запази целостта на съдържанието и защитата му от неоторизиран достъп;

Интегриране с външни системи: CMS обикновено е една от множество системи, използвани за представяне на информация в Интернет или интранет сайт. Затова възниква необходимост от интегрирането ѝ със съществуващите бизнес системи чрез отворени или индустриални стандарти;

Създаване на отчети: CMS да осигурява широк кръг от отчети за потребителите и администраторите. В най-добрия случай системата трябва активно да следи и да дава информация за всички възникващи събития и конфликти. Желателно е също отчетите да могат да се настройват.

3) Публикуване

Средствата за публикуване вземат съдържанието от хранилището и генерират крайните страници. Ключови изисквания от тази категория са:

- Стилови множества: Външният вид да се контролира чрез използването на стилови множества. Това осигурява гъвкавост и разширяемост;
- Шаблони на страниците: Цялостното оформление на страниците да се специфицира чрез темплейти. Добре е да е осигурен нетехнически интерфейс за тяхното управление;

- Разширяемост: Да има възможност за интегриране на отрязъци код за осигуряване на допълнителна функционалност на публикуването. CMS трябва да поддържа процес на непрестанни подобрения в дизайна на интерфейса;
- Поддържане на множество формати: CMS трябва да има възможност за публикуване на съдържание в множество формати като *HTML* (Hyper Text Markup Language), *PDF* (Portable Document Format), *WAP* (Wireless Application Protocol) и други. Добре е да има възможност за включване на поддръжка на допълнителни формати, които ще са необходими при появяването на нови стандарти. За да се постигне високо качество на всеки формат е жизнено важно да се разделя съдържанието от презентацията по време на създаването му. Това позволява да се използват различни стилови множества за всеки изходен формат;
- Персонализация: Възможност за предоставяне на различна информация въз основа или на потребителските профили или на мета данните на съдържанието. Това обикновено се изисква за големи портални уебсайтове;
- Статистики за използваемостта: CMS да поддържа пълни статистики за използваемостта на сайта, включително най-популярни страници, най-търсена информация, дневна посещаемост и др. Тази информация позволява да се следи успеха на сайта и да се идентифицират проблеми при използването му.

4) Презентация

Публикуваните страници трябва да отговарят на определени стандарти.

Ключови изисквания от тази категория са:

- Използваемост: Това обхваща аспекти като леснота на употреба и на усвояване. Използваемостта може да бъде установена чрез провеждане на тестове с реални потребители върху прототип на системата;
- Достъпност: CMS да отговаря на стандартите, като Инициативата за уеб достъпност на W3C (W3C Web Accessibility Initiative);

- Поддръжка на множество браузъри: Страниците да изглеждат по един и същи начин във всички по-използвани браузъри (Internet Explorer, Chrome, Firefox, Safari и др.).
- Ограничена функционалност от страна на клиента: Да се определи кои технологии от страна на клиента (Java, JavaScript, Flash и др.) да се използват. Това е по-важно за Интернет сайт, отколкото за интранет;
- Скорост: Размерът на страниците трябва бъде ограничен, за да се гарантира, че времето за зареждане е приемливо за потребителите с различни методи за достъп до Интернет (LAN, модем, кабелен и др.);
- Валиден HTML: Всички страници да отговарят на текущите HTML спецификации. Това осигурява максимална съвместимост с различните браузъри и платформи;
- Ефективна навигация: Да е осигурена логична, изчерпателна и лесна за използване навигация за потребителите;
- Метаданни: Всички страници трябва да осигуряват достатъчно мета данни за съдържанието. Това позволява ефективно индексирание и търсене от страна на търсещите машини;
- Приятелски URL адреси: Това са URL адреси от вида http://www.####/@@ @/****.htm и осигуряват правилното индексирание на уеб сайта от търсещите машини.

5) Бизнес изисквания

Бизнес изискванията на организацията също трябва да бъдат удовлетворени.

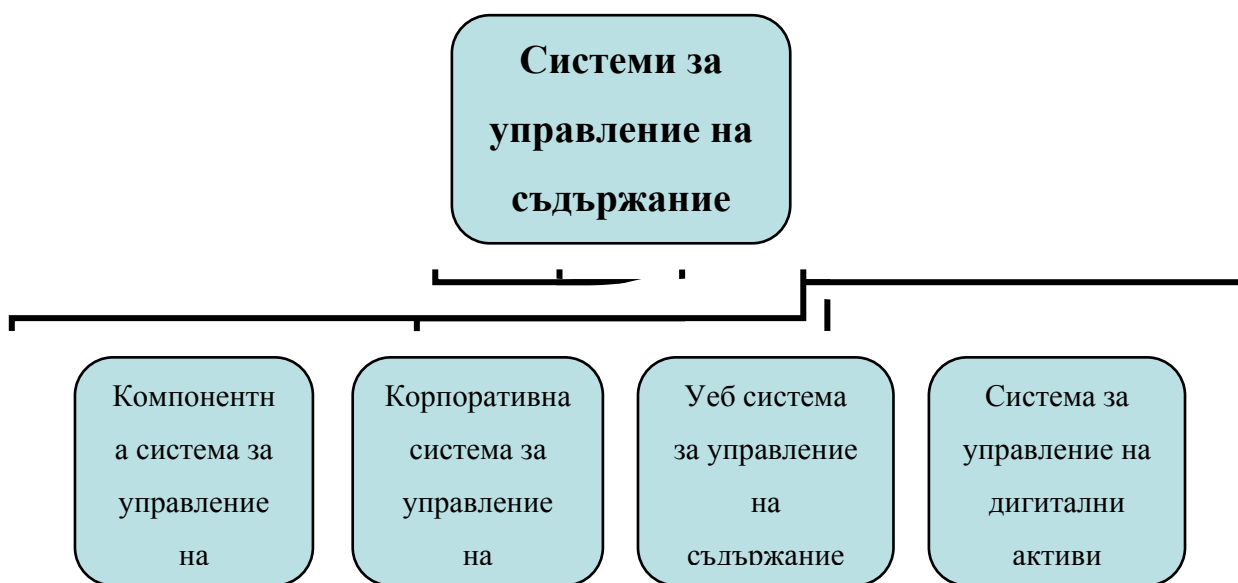
Ключови изисквания от тази категория са:

- Обучение: Производителят да е публикувал обучаващи материали за CMS и да може да осигури обучение за работа със системата;
- Документация: CMS да е съпроводена с пълна документация за потребители, администратори и разработчици;
- Гаранция: Да е осигурен гаранционен период за системата след нейната доставка;

- Поддръжка: Производителят да осигурява техническа поддръжка и обновяване на системата;
- Необходими ресурси: Хардуерът, софтуерът и операционната система, необходими за работа на CMS и интегрира ли се тя с вече съществуващите в организацията;
- Необходими умения: Какви знания и умения са необходими на служителите в организацията, които ще поддържат и настройват системата;
- Разходи: Какви фиксирани разходи за системата и разходи за всеки потребител са необходими. Второто е много важно за големи организации с множество служители;
- Съществуващи сайтове: Производителят да е осигурил списък на сайтове, където системата е била успешно имплементирана. Те могат да се сравнят с характеристиките и изискванията на организацията, където CMS ще се внедрява.

4. Основни видове системи за управление на съдържанието – предимства и недостатъци

Основните видове системи за управление на съдържанието могат да се видят на Фиг. 3. (на следващата страница), като всяка ще бъде разгледана накратко в следващите редове.



Фиг.3 Основни видове системи за управление на съдържанието

Компонентна система за управление на съдържанието

Компонентната система за управление на съдържанието е специализирана в създаването на документи от съставни части. Като например CCMS, която използва DITA (Darwin Information Typing Architecture) XML позволява на потребителите да сглобяват индивидуални теми в една карта (документ).¹³ Тези компоненти могат да се използват повторно с друг документ или с друго множество такива. Това гарантира, че съдържанието е наблюдавано и в целия набор документация. Все пак този вид система не е подходяща за големи организации, защото те имат възможност да си поръчат система, отговаряща на техните изисквания.

Корпоративна система за управление на съдържанието

Корпоративната система за управление на съдържанието организира документи, контакти и записи, свързани с процесите на търговската организация. Тази система структурира корпоративната информация по начин, който е най-подходящ за организацията, като предоставя лесна достъпност до нея от служителите и клиентите и осигурява максимална сигурност на информацията.

¹³ Shafie, H., DITA for the Impatient, Pixware SARL, Feb 24, 2018, p.18, 01.03.2018
<http://www.xmlmind.com/tutorials/DITA/DITA.pdf>

Концепцията ЕСМ е резултат от конвергенция на системи за:

- Управление на документи;
- Информацията на Web-сайтове;
- Автоматизация на делови процедури;
- Управление на знания и т.н.

ЕСМ се явява комплексно решение, осигуряващо:

- Събиране и систематизация на неструктурирана информация;
- Поддръжка на ориентирани към документи бизнес процеси;
- Възможности за анализ на информация.

На фигурата по-долу е представен жизненият цикъл на системата за управление на корпоративно съдържание.



Фиг. 4 Жизнен цикъл на системата за управление на корпоративно съдържание¹⁴

Система за управление на дигитални активи

Управлението на дигитални активи (УДА) се състои от задачи и решения, свързани с каталогизиране, съхранение, аотиране, извличане и разпространение на

¹⁴ Фигурата е адаптирана от Enter Enterprise Content Management and the Management of Content Life Cycle (Part 1), 21/11/2013, 07.03.2018 <http://smartdocumentmanagement.athento.com/2013/11/enterprise-content-management-and-the-management-of-content-life-cycle-part-one.html>

дигитални активи.¹⁵ Дигитални снимки, анимации, видео и музикални файлове представляват целевите зони на управлението на медийни активи. Системите за управление на дигитални активи (СУДА) съдържат в себе си софтуерни и хардуерни системи, които подпомагат процеса на управление на дигитални активи. Терминът „управление на дигитални активи“ се отнася и за протокола за сваляне, преименуване, създаване на копия, групиране, архивиране, оптимизиране, поддръжка и експорт на файлове.

При този вид системи управляваният „актив“ се взема и съхранява в цифров формат. Обикновено има целева версия или още наричана „същинска“ – версия на актива с най-висока резолюция и най-достоверно възпроизвеждане. Активът се описва чрез мета данни. Мета данните са описанието на самия актив, като дълбочината на описанието може да варира в зависимост от нуждите на системата, създателя или потребителя. Мета данните описват:¹⁶

- Съдържанието на актива (какво се съдържа в пакета);
- Начина на декодиране/кодиране (JPEG, tar (*tape archive*), MPEG 2);
- Произход – историята на актива до момента на „улавяне“;
- Собственост на актива;
- Право на достъп;
- Други.

Уеб система за управление на съдържанието

Уеб системата за управление на съдържанията е пакет или самостоятелно приложение за създаване, управление, съхраняване и използване на съдържанията в уеб страниците. Уеб съдържанието включва текст и вградени графики, снимки, видео, аудио файлове, както и програмен код. Уеб системата може да събира и индексира съдържания, избира и сглобява съдържания по време на работа или да доставя съдържания за определени потребители по определен начин, като например

¹⁵ Franziska Frey, Howard Vogl, Digital Asset Management— A Closer Look at the Literature, Printing Industry Center NY, March, 2005, p.5, 12.02.2018
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.369.7294&rep=rep1&type=pdf>

¹⁶ About Metadata, White Paper, 09/2004 p. 1, 12.02.2018
http://www.adobe.com/digitalimag/pdfs/about_metadata.pdf

друг език. Този вид системи обикновено позволяват на клиента да контролира HTML-базирани съдържания, файлове документи и уеб хостинг планове, въз основа на системата и дълбочината на нишата, която тя обслужва. *Този вид системи ще са обект на анализ в оставащата част от магистърската теза.*

Основният критерий, по който CMS (Web) могат да се разделят в според степента на откритост и софтуерния лиценз.

Степен на откритост - определя от правата за достъп до изходния код на програмата. По този критерий можем да диференцираме следните типове CMS:¹⁷

- CMS със затворен изходен код - потребителят няма достъп до кода. Повечето от комерсиалните системи се разпространяват по този начин. Друго име на този вид е патентована CMS.
- CMS с отворен изходен код – достъпът до изходния код на програмата се получава при определени условия. Има различни видове лицензи и условия, при които потребителят получава достъп до изходния код.
- CMS със свободен изходен код не само е с отворен код, но и е свободен. Последното означава, че потребителят получава изходния код на програмата и свободата да прави каквото поиска с изходния код и програмата. Потребителят получава следните права – изпълнява програмата за каквато и да е цел, да изучава как работи програмата и да я променя за своите нужди, да разпространява копия на програмата, да подобрява програмата и да публикува промените.

Софтуерен лиценз - CMS е софтуер, а той е обект на авторско право и обикновено отношенията между автора и крайния потребител се уреждат чрез специален договор, наричан софтуерен лиценз. Лицензите предоставят на потребителя правото да ползва CMS при определени условия.¹⁸ Предоставянето на това право обикновено става в рамките на няколко широко прилагани модела:

¹⁷ CMS Workbook, MichiganTech, May 2016, p.6, 12.02.2018 http://www.mtu.edu/mtu_resources/cms-workbook/cms-workbook.pdf

¹⁸ Classen, H., FUNDAMENTALS OF SOFTWARE LICENSING, IDEA: The Journal of Law and Technology 1996, p.355, 12.02.2018 <http://euro.ecom.cmu.edu/program/law/08-732/Transactions/Fundamentals.pdf>

- Платена CMS: Потребителят заплаща еднократно или периодично определена сума на автора на CMS. Това е най-често срещаният модел при CMS, разработвана със комерсиална цел.
- Безплатна CMS: Потребителят не заплаща за ползването на CMS.
 - Свободна CMS: Безплатна CMS с отворен код, която потребителят може да променя и препубликува при определени условия.

Почти всеки софтуер с отворен код е свободен софтуер. Двете понятия описват почти една и съща категория софтуер, но те стоят зад възгледи, базирани на коренно различни в основата си ценности, различните думи възплащават различни идеи. Отвореният код приема някои лицензи, които са недопустимо ограничаващи потребителите от гледна точка на свободния софтуер.¹⁹ Софтуерът с отворен код е софтуер, чийто изходен код е безплатен и публично достъпен, въпреки че специфичните лицензионни споразумения варират в зависимост от това какво е позволено да се прави с този код.

В софтуера със затворен код (патентования софтуер) често срещани са злонамерени функционалности като шпиониране на потребителите, ограничаването им, задни врати и принудителна актуализация, а някои от поддръжниците на отворения код искат да ги реализират и в софтуера с отворен код.²⁰ Под натиска на различни компании, софтуерът все по-често се проектира специално, за да ограничава правата на ползващите го. Следват предимствата и недостатъците на системите за управление на съдържанието с отворен и затворен код:

Предимствата на CMS със затворен изходен код са:

- по-голяма сигурност срещу хакерски атаки и кражба на изходния код, поради това че, когато изходния код не е публично достъпен, обикновено сайта не е податлив на хакерска атака “defacements”, която се използва за промяна на външния вид на сайта;

¹⁹ Вж. по-горе.

²⁰ Geisert, S., Railsback, S., Content Management System User Guide, COLORADO STATE UNIVERSITY, 2008, pp.31-35, 12.02.2018 https://code.colostate.edu/wp-content/uploads/sites/17/2016/11/CMS_Manual_v2.pdf

- поддръжка – обикновено има екип от хора, които са на разположение. Получава се качествена и бърза поддръжка по телефон, имейл или в специализирани форуми;
- настройки – понякога се предлагат възможности и функции, които не са налични в отворения код;
- пълна и лесна за употреба документация за различни нива потребители;
- описание на приложно-програмният интерфейс (API).

Основни недостатъци на CMS със затворен изходен код са:

- липсата на гъвкавост за промяна;
- затворена система – не може да преместите сайта и софтуера при друг разработчик;
- зависимост от разработчиците, които притежават кода – в случаите когато нещо не е наред със сайта, за създаването на подобрения и нови функции, за поправянето на бъгове;
- цена на поддръжката – плаща се за всяка извършена услуга;
- хостинг разходи – може да изисква употребата на патентовани хостинг услуги, като Microsoft Office с IIS и NET.
- липса на актуални помощни материали на разположение за някои от съществуващите CMS.

Предимствата на CMS със свободен изходен код са:

- лесна адаптация на функционалността, предлагат повече функции;
- гъвкавост – със свободния код може да се направи всичко, да се добавя и променя;
- общността на потребителите: поради широкото разпространение на CMS с отворен код те имат по-голяма база от потребители и повече хора работят върху софтуера, поправят бъгове, създават модули, създават нови и по-добри версии;
- големия брой разработчици гарантира високо качество на кода;

- голям брой независими компании, предлагащи услуги по внедряване и поддържане на системата.

Сред недостатъците са:

- лесно намиране на пропуски в сигурността, ако кода на сайта се знае. Това отваря потенциалните рискове за сигурността и уязвимости;
- съществуващите лицензионни ограничения върху използването на определени формати за данни в системите с отворен код;
- често липсва обучение и сертифициране на специалисти, участващи в изпълнението и техническа поддръжка на системата;
- няма директна поддръжка – системата, няма телефонен номер на който да се обади човек, когато нещо се развали на сайта;
- документация – има проекти с отворен код в Интернет, които имат наистина голяма документация и такива, които са без никаква, така че документацията е или много добра или е пропуск.

ГЛАВА II. ХАРАКТЕРИСТИКА И СРАВНИТЕЛЕН АНАЛИЗ НА СИСТЕМИТЕ ЗА УПРАВЛЕНИЕ НА СЪДЪРЖАНИЕ С ОТВОРЕН КОД – DRUPAL, WORDPRESS, JOOMLA!

1. Преглед на Drupal, WordPress, Joomla!

Напоследък безплатните и свободни CMS с отворен код като Drupal²¹, Joomla!²² и WordPress²³ се превръщат в голяма конкуренция на патентованите CMS. Причините за това са, че те са лесни за използване и не се нуждаят от умения за програмиране, за да се правят актуализации или промени. *По-голямата част от информацията във тази точка е взета от официалните уеб страници на гореспоменатите CMS.*

1.1 Drupal

Drupal е система за управление на съдържанието, написана на езика PHP и използваща в качеството на съхранение на данни релационна база данни. *Drupal* е с отворено съдържание, предоставена с лиценза за свободна документация на GNU General Public License|GPL, което означава, че може да се сваля и използва свободно, както и да се споделя с други. Този отворен развоен модел означава, че потребителите могат постоянно да развиват системата, за да осигурят едно от първите места на проекта, който използва най-новите интернет технологии.

Drupal е система за изграждане на сайт и управление на съдържанието, с помощта на която може бързо и без специални познания по уеб дизайн и програмиране да се поддържа даден уебсайт. След инсталацията на *Drupal* на хостинг сървър, цялата администрация на съдържанието на сайта се осъществява онлайн от административната секция и промените се отразяват незабавно на сайта. *Drupal* е подходящ за изграждане на личен или фирмен сайт с блог и форум. Има отлични модули и за електронен магазин. Вградените модули дават възможност за

²¹ <https://www.drupal.org/>

²² <https://www.joomla.org/>

²³ <https://wordpress.org/>

подредба на съдържанието по категории, оптимизация за търсачки, публикуване на новини от сайта ви в други сайтове с RSS (Rich Site Summary) и още много полезни функции.

История на проекта

През 2000 г. белгиецът Дрис Бейтарт (*Dries Buytaert*) започва разработката като ръководител на проекта. През 2000 г. група студенти от университета в Антверпен използвали общ ADSL-модем за достъп до Интернет и Дрис Бейтарт създал вътрешен сайт за общуване в тази малка локална мрежа. Програмата, на която работел сайта, нямала име затова Дрис решил да регистрира домейн и в резултат на грешка вместо `door.org` регистрирал `dror.org` Аудиторията на сайта с времето се променила към интересуващи се от новите уеб технологии и новия сайт се превърнал от любителски експеримент в проект обслужващ идеите на уеб програмирането. През 2001 г. Дрис публикувал изходния код на `dror.org` под названието „Drupal“. От тогава разработката е преместена на по-късно регистрирания сайт `drupal.org`, който придобил на официален сайт на проекта.²⁴

Официалните разработчици на ядрото сега са десетки, но много потребители изпращат пачове включени в кода на системата. Така функционалността се разширява с допълнителни модули, разработени от различни разработчици. През 2011 г. са разработени повече от 7000 допълнителни модула, разширяващи функционалността на CMS. През 2007 г. Дрис Бейтарт създава фирмата *Acquia*, за комерсиална поддръжка на проекти и услуги, базиращи се на *Drupal* и развили бизнеса си за поддръжка на отворени проекти. Постепенно с течение на времето се появили национални общества. Вече работят повече от 30 национални сайта поддържани на различни езици²⁵.

Развитие на Drupal

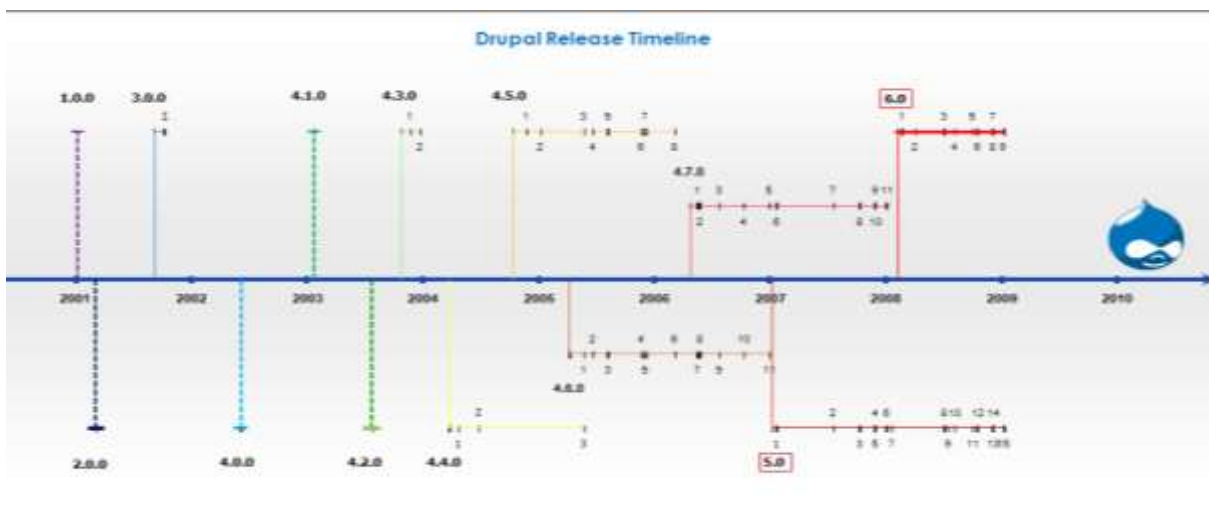
Интересът към *Drupal* значително се увеличава през 2003 г., когато е използвана за направата на страницата „DeanSpace“ за Howard Dean, кандидат за

²⁴ 12.02.2018 <https://www.drupal.org/about/history>

²⁵ 12.02.2018 <https://assoc.drupal.org/about/history>

президент през 2004 г. от Демократическата пария. DeanSpace използва свободното споделяне на *Drupal*, за да поддържа около 50 различни неофициални уебсайта, които комуникирали помежду си и с организацията на изборите. След приключване на кампанията на Dean, членовете на неговия уеб екип продължили да развиват платформата, която можела да подпомага политическата активност чрез създаването през 2004 г. на CivicSpace Labs, първата компания със служители на пълен работен ден, развиващи и разширяващи технологията на *Drupal*. Започнали и други компании да се специализират в развитието на *Drupal*.²⁶ Към 2013 г. има стотици уебсайтове предлагащи *Drupal* произлизащи технологии.

На 5 март 2009 г. Дрис Бейтарт анонсира code freeze за *Drupal 7* от 1 септември 2009 г. *Drupal 7* е представена на 5 януари 2011 г. в няколко държави. След това се прекратява поддръжката на *Drupal 5*, като тя остава само за *Drupal 6* и *Drupal 7*. Последната версия е *Drupal 8.50* предоставена на 7 март 2018г. Работата по *Drupal 8* е разделена на категории наречени Core initiatives: Mobile, Layouts, Web Services, Configuration Management, and HTML5. За да се онагледя процеса на разработване и усъвършенстване на *Drupal*, на Фиг. 5 е показана графика, изобразяваща развитието на системата и нейните версии във времето от годината на създаването ѝ.



Фиг. 5 Развитие на Drupal 1-6 през годините²⁷

²⁶ Вж. по-горе.

²⁷ 12.02.2018 https://en.wikipedia.org/wiki/Drupal#/media/File:Drupal_release_timeline.png

Днес *Drupal* се развива от общността и популярността ѝ нараства бързо. От юли 2016 г. до юни 2017 г., *Drupal* е изтеглена повече от 1,4 милиона пъти от официалния сайт Drupal.org, което е покачване от 125% в сравнение с предходната година.²⁸ В края на 2016 г. повече от 980 000 сайтове използват *Drupal*. Това включва стотици добре познати организации, включително медийни и издателски компании, правителства, нестопански организации, училища и личности.

Принципи, заложиени в Drupal

- *Модулност и разширяемост.* Drupal се стреми да предостави малко, мощно ядро предоставящо лесно разширяване чрез потребителски модули.
- *Качествено кодиране.* Високо качествения, елегантен, документиран код е приоритет пред грубата функционалност.
- *Вградени стандарти.* Drupal поддържа установени и нови стандарти.
- *Ниски ресурсни изисквания.* За да осигури висока производителност, Drupal се стреми към ниско кодиране на профила (например чрез минимизиране на заявките от базите данни). Drupal също така има минимални, широко достъпни изисквания за софтуера на сървъра. Конкретно Drupal трябва да е изцяло работеща на платформа с уеб сървър, PHP, и една от MySQL или PostgreSQL.
- *Отворено съдържание.* Drupal е основана на философията за отворено съдържание, свободен софтуер и лиценз на GNU. Конкретно Drupal е кодиран на скриптовия език с отворено съдържание PHP и поддържа основно данни с отворено съдържание като MySQL and PostgreSQL.
- *Леснота при използване.* Drupal се стреми към висок стандарт на използване от разработчици, администратори и потребители.

Ядро на Drupal

В общността на *Drupal*, под понятието ядро се разбира нещо извън директорията на сайта в инсталацията на *Drupal*. В базовата конфигурация, съдържанието на уебсайт на *Drupal* може да се развива от регистриран или

²⁸ 12.02.2018 <https://websiteworker.com/drupal-history-stats-and-facts>

анонимен потребител (при разрешение от администратора) и е достъпен до веб-потребители чрез различни критерии за избор. Ядрото на *Drupal* включва йерархична система, която позволява съдържанието да бъде категоризирано или маркирани с ключови думи за по-лесен достъп.

Drupal има незадължителни модули, които могат да бъдат разрешени от администратора, за да се разшири функционалността на ядрото на уебсайта. *Drupal* може автоматично да предупреждава администратора за нови версии на модули, теми или ядрото. Дистрибуциите на ядрото на *Drupal* предоставят много свойства, включващи:

- *Access statistics and logging* – статистика на достъпа и логването в системата;
- *Advanced search* – разширено търсене;
- *Blogs, books, comments, forums, and polls* – блогове, книги, коментари, форуми и анкети;
- *Caching and feature throttling for improved performance* – кеширане на характеристиките с цел подобряване на производителността;
- *Descriptive URLs (Uniform Resource Locator)* – описателни глобални уеб адреси;
- *Multi-level menu system* – системно меню на много нива;
- *Multi-site support* – поддръжка на множество уебсайтове едновременно;
- *Multi-user content creation and editing* – създаване и редактиране на съдържание от много потребители;
- *OpenID support* – поддръжка на свободно логване в системата;
- *RSS feed and feed aggregator* – формат за запазване на информация от глобалната мрежа;
- *Security and new release update notification* – уведомления, свързани със сигурността и обновяването;
- *User profiles* – потребителски профили;

- *Various access control restrictions (user roles, IP addresses, email)* – различни ограничения за достъп (роли на потребителите, IP адреси, имейли);
- *Workflow tools (triggers and actions)* – инструменти за процеса на работа (тригери, действия).

В *Drupal* е предоставена гъвкава схема за организация на структурата на сайтове на основа на *таксиномия* (механизъм, позволяващ създаването на произволно количество тематични категории за съдържанието на сайт и асоциирането им с модули, обезпечавщи входа и изход на информация; категориите могат да бъдат обикновен или йерархически списък, с произволно сложна структура, където елементите могат да няколко „родителски“ и „дъщерни“ елементи²⁹). С помощта на тази схема и модули е възможно организирането на различни варианти структурирано съдържание. Например лесно се създава списък по ключова дума за всички документи в сайта.

Модули включени в Drupal

Най-често използваните модули, които могат да се изтеглят от хранилището на drupal.org са:

- *Views* - модул с интерфейс за създаване на различни списъци;
- *Panels* - предоставя възможност за разбиване съдържанието на страница на региони;
- *Service* - създаване на собствени уеб услуги;
- *WYSIWYG* - модул за работа с wysiwyg редактори, позволява едновременно използване на няколко редактора (ckeditor, tinymce, whyzziwyg);
- *Panels everywhere* - предоставя възможност за използване на главния панел за целия сайт, с последващото и вкарване на други панели;
- *Features* - позволява сливане на определени бази данни в кода във вид на модули с последващ трансфер;

²⁹ 12.02.2018 <https://www.drupal.org/about/mission-and-principles>

- *Backup and migrate (BAM)* - възможност за дъмп и възстановяване на базата напълно или частично;
- *Views bulk operations* - групови операции над обекти в списъци;
- *Devel* - набор средства и функции за бърза разработка;
- *Drush* - Drupal shell, пакет за работа с Drupal и модули чрез конзолата;
- *Memcache* - интеграция със сървъри memcached;
- *Ubercart* - система за електронна търговия.

1.2 WordPress

WordPress е вид свободен софтуер с отворен код за създаване на блогове и/или сайтове, използващи система за управление на съдържанието и е базиран на PHP и MySQL.

Системата може да се ползва по два начина: или директно в Интернет, например в сайта WordPress.com, или за индивидуална употреба - WordPress пакета се инсталира на собствен компютър и той изпълнява функциите на сървър. *WordPress* предлага изключително лесен и интуитивен административен панел, който е полезен за обикновените потребители. Същевременно отвореният код предлага голяма свобода на специалистите. Цялата система предлага огромен набор от безплатни и (относително) евтини шаблони, наречени „теми“ (themes) и разширения (plugins).

История на WordPress

През юни 2001 г. Мишел Валдриги започва работа върху „енджина“ *b2/cafeblog*. Той се счита за предшественик на WordPress и бива инсталиран приблизително на 2000 блога до май 2003 г. През същата година към проекта се присъединяват Мат Муленвег и Майк Литъл и създават WordPress.³⁰ Информация от януари 2015 г. сочи, че повече от 23,3% от най-популярните 10 милиона уебсайтове сега използват WordPress. Към 31 декември 2017г. WordPress е

³⁰ 12.02.2018 <https://codex.wordpress.org/History>

използван от 60,2% от всички уебсайтове, чийто управление на съдържание ни е известно. Това са 30% от всички уебсайтове.³¹

Високата популярност се дължи на това, че WordPress предлага:

1. Възможност за публикуване с помощта на програми от трети страни;
2. Лесно инсталиране и конфигуриране;
3. Поддръжка на RSS, Atom, trackback, pingback;
4. Подключаеми разширения (plugins) с уникална проста система за взаимодействие с кода; поддръжка на така наречените „теми“, позволяваща лесна промяна, както на външния вид, така и на начина на представяне на данни;
5. „Темите“ са реализирани като набор от файлове-шаблони, което влияе положително на скоростта и гъвкавостта;
6. Огромни библиотеки от „теми“ и „плъгини“;
7. Зададеният архитектурен потенциал позволява лесното реализиране на комплексни решения.

Функционалност

WordPress притежава web-базирана система от шаблони, които използват т.нар. шаблонен процесор. Платформата предлага огромен набор от „теми“, които придават различен облик и функционалност на даден блог. Различните теми не променят съдържанието и не нарушават целостта на сайта. Част от тях са платени, а друга са със свободен или ограничен достъп. Освен това съществува възможност потребителят сам да създава и развива свои собствени теми.

Всеки WordPress сайт изисква избор на поне една *тема*. Всяка тема трябва да бъде създадена съгласно стандартите на WordPress, използвайки структуриран PHP и валиден HTML. С цел подобряване на функционалността, PHP и HTML кодът в темите може да бъде добавян или променян. Системата предлага две различни възможности за инсталирането на дадена тема:

³¹ Usage statistics and market share of WordPress for websites, Technological Survey, 07.03.2018
<https://w3techs.com/technologies/details/cm-wordpress/all/all>

1. през самата система;
2. чрез FTP (file transfer protocol).

Архитектурата на WordPress използва *разширения* (plugins). До момента (2018 г.) съществуват 54 453 различни добавки, всяка от които предлага собствени функции и характерни черти.³² Разширената функционалност позволява на потребителите да приспособят сайта си към своите специфични нужди. Тези персонализации варират от оптимизация за търсачки до клиентски портали (client portals), използвани за показване на лична информация при влизане в профила на потребителите, както и системи за управление на съдържанието, или функции за показване на съдържанието като например добавяне на джаджи (widgets) и навигационни ленти (navigation bars). Трябва да се има в предвид, че не всички налични разширения са съвместими с последната налична версия на WordPress и поради това тези приставки може или да не функционират правилно, или изобщо да не работят. Разширенията са с отворен код и могат да бъдат свободно променяни от програмистите. Голяма част от тях са безплатни, а други са платени или с ограничен достъп.

WordPress също така предлага интегрирано управление на връзките (integrated link management); „приятелски” търсачки (search engine-friendly); структура, базирана на постоянна връзка (препратка); възможност да се присвоят няколко категории към статиите; поддръжка за обозначаване на етикети на постове и статии. Автоматичните филтри осигуряват стандартизирано форматиране и оформяне на предварително зададен стил на текста в статиите.

WordPress поддържа стандартите Trackback и Pingback за показване на връзки към други сайтове, които сайтове от своя страна са свързани към пост или статия. Постовете в блог, създаден в WordPress, могат да бъдат редактирани в HTML с помощта на визуален редактор или с помощта на определен брой разширения, които осигуряват разнообразие от персонализирани функции за редактиране. До трета версия WordPress поддържа един блог за една инсталация,

³² Вж. по горе.

въпреки че няколко копия могат да се стартират от различни директории, ако се конфигурират да използват отделни таблици в базата данни.

Разработка и поддръжка

Мат Муленвег и Майк Литъл са съоснователите на проекта. Основните разработчици са Хелън Хю-Санди, Дион Хулс, Марк Якуит, Мат Муленвег, Андрю Оз и Андрю Накин.³³ WordPress е разработен също от неговата общност, включително тестери, група от доброволци, които тестват всяка версия. Още в най-ранен стадий те имат достъп до изграждането на неутралния софтуер, бета версиите и версиите в процес на разработка. Грешките се документират в специален “mailing list”.

Въпреки че, до голяма степен WordPress се разработва от свързаната с него общност, WordPress е тясно свързан с Automattic - компания, основана от Мат Муленвег. На 9 септември 2010 г. Automattic предава търговската марка WordPress на новосъздадената WordPress Foundation, която е основна организация поддържаща Wordpress.org (включително софтуерът и архивите за разширенията и темите)³⁴.

Основният сайт за поддръжка на WordPress е Wordpress.org. Той поддържа on-line ръководството WordPress Codex, което е и постоянно обновяващо се хранилище за информация и документация за WordPress, и WordPress Forums – активна on-line общност за WordPress потребители.

1.3 Joomla!

Joomla! е безплатна система за управление на съдържанието с отворен код, написана на PHP, за публикуване на уеб съдържание. Използва база данни MySQL и техники на обектно-ориентирано програмиране. Joomla! включва/съдържа възможности като кеширане на страници (page caching) за подобряване на изпълнението, RSS четци, версии на страниците за печат, предаване на кратки новини (т.нар. "news flashes"), блогове, анкети (web polls), уебсайт търсене

³³ McKeown, S., Wordpress – Freedom, Community and the Business of Open Source, p.2, 12.02.2018
<https://wordpress.org/about/history/chapter3.pdf>

³⁴ Вж. по-горе, р. 31

и езикова локализация. Към юли 2013 г., Joomla! е била изтеглена над 35 милиона пъти. Над 6000 безплатни, некомерсиални разширения са достъпни от официалната страница на Joomla! Смята се, че тя е втората най-използвана CMS след WordPress. Joomla! има редица преводи на над 60 езика, измежду които и български. Глобалната общност от потребители и разработчици се състои от над 150 хиляди души.³⁵

Подобно на много други популярни уеб приложения, Joomla! може да се изпълнява на LAMP (безплатен пакет софтуер, с отворен код). Много хостинг доставчици имат контролни панели, които позволяват автоматичната инсталация на Joomla!. На Windows, Joomla! може да се инсталира с помощта на Microsoft Web Platform Installer, който автоматично разпознава и инсталира всичко липсващо, като например PHP или MySQL. Joomla! се радва на силна подкрепа от хостинг доставчици, които се специализират в нея или могат да я поддържат. Много уебсайтове предоставят информация за инсталиране и поддръжка на Joomla! сайтове.

История на Joomla!

Joomla! е резултат от отцепването на екип от Mambo на 17 август, 2005 г. , който взема сорс кода и започва самостоятелна разработка върху него. Разработчиците от Joomla! създават уебсайт, наречен OpenSourceMatters.org (OSM), за да разпространяват информация за потребителите, разработчици, уеб дизайнери и за общността като цяло. Ръководител на проекта Андрю Еди (Andrew Eddie) написва писмо, което се появява в частта за известия на обществен форум mamboserver.com. В рамките на един ден малко повече от хиляда души се присъединили към OpenSourceMatters.org повечето, от които изразявали своята подкрепа и насърчавали проекта, в резултат на което сайтът бил претоварен и блокирал.³⁶

³⁵ Honeyager, J., History of Joomla, Article, 07.03.2018 <http://expertwebprofessionals.com/news-data-blogs-web-experts/articles-about-joomla/joomla-background/103-history-of-joomla.html>

³⁶ Вж. по-горе.

През първите две седмици след съобщението на Андрю Еди, екипите биват реорганизирани и Общността продължава да расте. Андрю Еди призовава общността да избере име за проекта от списък с предварително избрани имена. Основният екип посочил, че ще вземе окончателното решение за името на проекта в зависимост от решението на общността. В крайна сметка основният екип избира име, което не е в списъка с предложени имена. На 22 септември 2005 г. новото име „Joomla!“, бива обявено. Това е английското произнасяна на думата от езика Суахили Jumla, която означава „всички заедно“ или „като едно цяло“. На 26 септември 2005 г. екипът разработчици представя лого и приканва Общността за гласуване по него.³⁷

Версии на Joomla!

- Joomla! 1.0 е пусната на 22 септември 2005 г. въз основа на Mambo 4.5.2.3.
- Joomla! 1.5 е пусната на 22 януари, 2008 г. Joomla 1.6 е пусната на 10 януари, 2011 г.
- Joomla! 1.7 е пусната на 19 юли 2011 г., шест месеца след 1.6.0. Тази версия добавя по-голяма сигурност и по-добри инструменти за миграция.
- Joomla! 2.5 е пусната на 24 януари 2012 г., шест месеца след 1.7.0.
- Joomla! 3.0 е пусната на 27 септември 2012 г.
- Joomla! 3.1 е пусната до 24 април 2013 г.
- Joomla! 3.8 – последната стабилна версия

Шаблони за Joomla!

Шаблоните за Joomla! са многостранни разширения, които са отговорни за оформлението, дизайна и структурата на задвижваните от Joomla! сайтове. Докато CMS управлява съдържанието, шаблоните управляват външния вид, възприемането на елементите на това съдържание и цялостния дизайн на задвижваните от Joomla! сайтове. Съдържанието и дизайна в шаблоните на Joomla!

³⁷ 01.03.2018 <https://www.joomla.org/announcements/release-news/5654-joomla-3-5-is-here.html>

са отделни и могат да се редактират, променят и изтривани поотделно. Шаблонът е мястото, където е зададен дизайна на основният *layout* за сайт на Joomla!. Това включва, мястото където потребителите поставят различни елементи (компоненти, модули и плъгини), които са отговорни за различните видове съдържание. Ако шаблонът може да бъде персонализиран от потребителя, потребителят може да промени разположението на съдържанието в сайта, например, да постави главното меню от дясната или лявата страна на екрана. Има много платени и безплатни шаблони на разположение в Интернет.

Шаблонът е мястото, където е зададен дизайна на главното оформление за един Joomla! сайт. Това включва мястото, където потребителите поставят различни елементи (компоненти, модули и плъгини), които са отговорни за различните видове съдържание. Компонентите на шаблоните са:

- *Цветова схема* - използвайки CSS (Cascading Style Sheets) в дизайна на шаблона, потребителите могат да променят цветовете на фона, текст, линкове или почти всичко, което те биха могли да променят използвайки HTML.
- *Снимки и ефекти* - потребителите могат също да контролират начина, по който изображенията се извеждат на страницата и дори да създават флаш ефекти като падащи менюта.
- *Шрифтове* - дизайнът за шрифтовете е заложен в CSS файла на темплейта, създавайки единен изглед за целия сайт, което прави лесно промяната на целия изглед само чрез промяна на един или два файла, а не всяка една страница по отделно.

Разширенията за Joomla! спомагат за разширяването на възможностите на Joomla! сайтовете. Има пет вида разширения: компоненти, модули, плъгини, шаблони и езици. Всяко едно от тези разширения управлява конкретна функция. Нека да ги разгледаме накратко:

Компоненти

Това е основно средство, което разширява функционалните възможности на Joomla! и представлява набор от скриптове, които отговарят за определени

функции. В повечето случаи компонентите се виждат в централната колона на сайта. Компоненти в Joomla! се явяват форуми, файлови архиви, галерии, системи за статистика, и т.н. Управление на потребителите, създаване на категории, – всичко това се изпълнява от компоненти. Компонентите в Joomla, се състоят от две части: административен панел на компонента и фронтенд (сайт). Една от частите може да отсъства, например стандартния компонент Search няма административна част, а компонента Config (настройка на глобалните параметри в сайта), съдържа само административна част.

Плъгини

Това са по-напреднали разширения и по същество, обработват дадено събитие. По време на изпълнение на част от Joomla!, на модул или компонент, събитие може да бъде задействано. Когато събитие се задейства, плъгина може да бъде извикан, за да се справи с това събитие. Например, плъгин може да блокира потребителски статии или да филтрира лоши думи.

Шаблони

Описват основния дизайн на сайтовете в Joomla!. Те са разширенията, които позволяват на потребителите да променят външния облик на сайта. Потребителите ще виждат модули и компоненти в шаблона. Те са адаптивни и гъвкави. Шаблоните определят „стила“ на даден сайт.

❖ Модули

Средство за разширяване на възможностите в Joomla!. В повечето случаи, модулите изпълняват функции, които показват информация и се явяват допълнение към компоненти. Модулите се разполагат в ляво или дясно на сайта. Примерни модули са: навигационното меню, календари, списък с популярни статии и т.н. При публикуване на модула може да укажете на каква позиция в шаблона да се намира той.

Езици

Много прости разширения, които може да се използват като основна част или като софтуерно разширение.

В следващата част от Глава II ще се направи подробно сравнение на трите системи за управление на съдържанието – Drupal, WordPress и Joomla!.

2.Сравнение на системите за управление на съдържанието с отворен код – Drupal, WordPress и Joomla!

Сравнението ще бъде направено по няколко критерия – инсталационно време и сложност, персонализиране и възможност за обновяване, сигурност. След това ще бъде направен обобщен анализ на резултатите с помощта на графики.

Инсталация и сложност на процеса

Инсталацията е първото нещо, което трябва да се направи, преди да се започне с използването на CMS, така че нека да разгледаме какво е необходимо, за да инсталирате тези инструменти (Drupal, Joomla!, WordPress).

Drupal

Drupal е смятан от мнозина за най-сложните за инсталиране и използване от трите посочени по-горе, но това не е вече така. Drupal се е развил и процесът е доста по-прост. Инсталационните файлове се свалят от интернет, разархивират се и съдържанието се поставя в главната (root) папка на веб сървъра. След това главната папка се отваря чрез браузъра и софтуера започва сам да извършва процеса. Важно е да не се забравя да се създаде база от данни за Drupal сайта и потребителското има и парола да са лесно достъпни, преди да започне процеса на инсталация.

Joomla!

Подобно на Drupal, Joomla! също се нуждае от предоставяне на името на базата от данни по време на инсталацията. В процеса на инсталиране на Joomla! е подобен на Drupal с изключение на няколко допълнителни опции, които Joomla! осигурява по време на инсталацията. Например, може да се избере дали Joomla!

сайта трябва да бъде онлайн след инсталацията. Преди окончателното приключване на инсталацията, трябва да се направят всички необходими конфигурации. Също така, като защитна функция, инсталаторът изисква премахването на инсталационната папка с кода след инсталацията.

WordPress

Смята се, че WordPress е най-лесната CMS за използване от трите разглеждани. WordPress изисква същата информация като другите две, но това е добре скрито зад два етапа на инсталация. Първата част е създаването на config.php файл (цялата информация за базата от данни, потребителско име / парола и др. се намира във файла). Когато това се направи, WordPress се инсталира само с едно потвърждение. Ако има вече готов файл config.php (от предишното инсталиране или ако ръчно е създаден) не е необходимо да се прави първата стъпка. Инсталаторът автоматично търси файла и ще изисква създаването на config.php файл само ако такъв не е налице.

Леснота на употреба

Леснотата на употреба е друг важен аспект на CMS. Те имат много функции, а тяхното изучаване с ръководството за употреба отнема много време. Ето защо дали една CMS е лесна за използване, или не играе огромна роля за нейното приложение от страна на потребителите.

Drupal

Drupal предоставя някои много важни функции в един основен семпъл потребителски интерфейс (UI). След като се влезе в администраторския профил, в горната част има лента с менюта, показваща всички важни аспекти на Drupal сайта. Има връзка за съдържание, която показва списък на цялото съдържание и коментарите в сайта и позволява да се добавят нови или съществуващите да се управляват (например, публикуване/премахване). Другите връзки в менюто също са интуитивни: Структура (Structure), Външен вид (Appearance), Хора (People), Модули (Modules), Конфигурации (Configurations), Отчети (Reports).

Joomla!

Администраторската страница на Joomla има прекалено много менюта, разположени вертикално и хоризонтално, което отначало затруднява работата със системата. Но всъщност менюто в лявата част на страницата е просто извадка на най-важните връзки от главното меню в горния край. Както и с Drupal, Joomla! изброява всички основни аспекти на сайта като различни елементи от менюто и всеки елемент е с падащо меню, предоставящо допълнителни връзки. Като цяло администраторският интерфейс на Joomla! е по-прегледен (в сравнение с Drupal) и също така осигурява възможност за по-фини настройки на контрола върху сайта. Като недостатък може да се посочи наличието на твърде много бутони и връзки по цялата работна площ, което може да затрудни тяхното оптимално използване, без да се прегледа документацията.

WordPress

Интерфейсът на WordPress е минималистичен и използва лесен за разбиране език, което отличава системата от другите две. В сравнение с интерфейса на Joomla! и Drupal, които използват по-технически език, WordPress определено има предимство тук. За уебсайтове, управлявани от потребители с малък или никакъв технически опит, или за малки уебсайтове, при които са необходими с чести актуализации, WordPress е най-подходящата CMS (основно поради изчистения интерфейс). Все пак предимство на Joomla! са много настройки и контроли, които позволяват да се управлява сайта в по-голяма степен. Drupal предоставя почти същите възможности, но с по-прегледен и лесен за употреба интерфейс.

Персонализиране и актуализация

Как може да се персонализира и надгради една CMS е друг важен аспект при нейното използване. С течение на времето, всяка CMS трябва да бъде осъвременявана с цел по-добра функционалност, сигурност или друга причина. Също така, много пъти темите или пългините, които се предоставят по подразбиране не са точно това, което потребителите желаят. Въпреки че

персонализирането изисква определено ниво на техническа грамотност, потребителският опит е това, което прави разликата между различните CMS.

Drupal

Единственият начин за ъпгрейд на инсталацията при Drupal е той да се направи ръчно, т.е. архивират се старите файлове и данни, изтегля се най-новият пакет на Drupal и всички стари файлове се заменят, освен папката със сайтове (съдържа теми и други данни). Това може да звучи като трудна задача за потребители без опит в областта и има известна степен на риск, ако нещо се обърка целят сайт да се изгуби.

Joomla!

Joomla! поддържа обновяване на ядрото чрез компонента за актуализация на Joomla! (версия $\geq 2.5.4$) или от раздел Актуализация (Update) в Joomla Extension Manager (версия $< 2.5.4$). Въпреки това, в някои случаи, този метод актуализация не може да се използва. Други методи за актуализиране на Joomla! са: инсталационен метод, където се избира файл за актуализация; ръчна актуализация, където файловете трябва да се заменят ръчно. Що се отнася за редактиране на теми, те трябва да се редактират офлайн или чрез инсталиране на допълнителен плъгин.

WordPress

Подобно на Joomla!, WordPress също поддържа онлайн актуализации чрез администраторския потребителски интерфейс. WordPress има функцията да известява потребителя, когато има налична актуализация. Съществува и възможност за ръчно актуализиране. Друга интересна особеност е онлайн редактирането на файлове. То позволява да се персонализират темите или плъгините чрез редактиране на файловете в самото приложение.

Сигурност

Сигурността е много важен фактор при избора на CMS за даден уебсайт. Почти всеки уебсайт в интернет е уязвим откъм заплахи за сигурността.

WordPress

WordPress е създадена на базата на много сигурен код и отговаря на хакерски атаки много бързо. WordPress също има механизъм за автоматично обновяване, което позволява на WordPress сайтовете да се актуализират автоматично, когато са налице нови поправки в сигурността. WordPress сайтовете могат да бъдат допълнително защитени с автоматични архиви, двуфакторна автентификация и други способности.

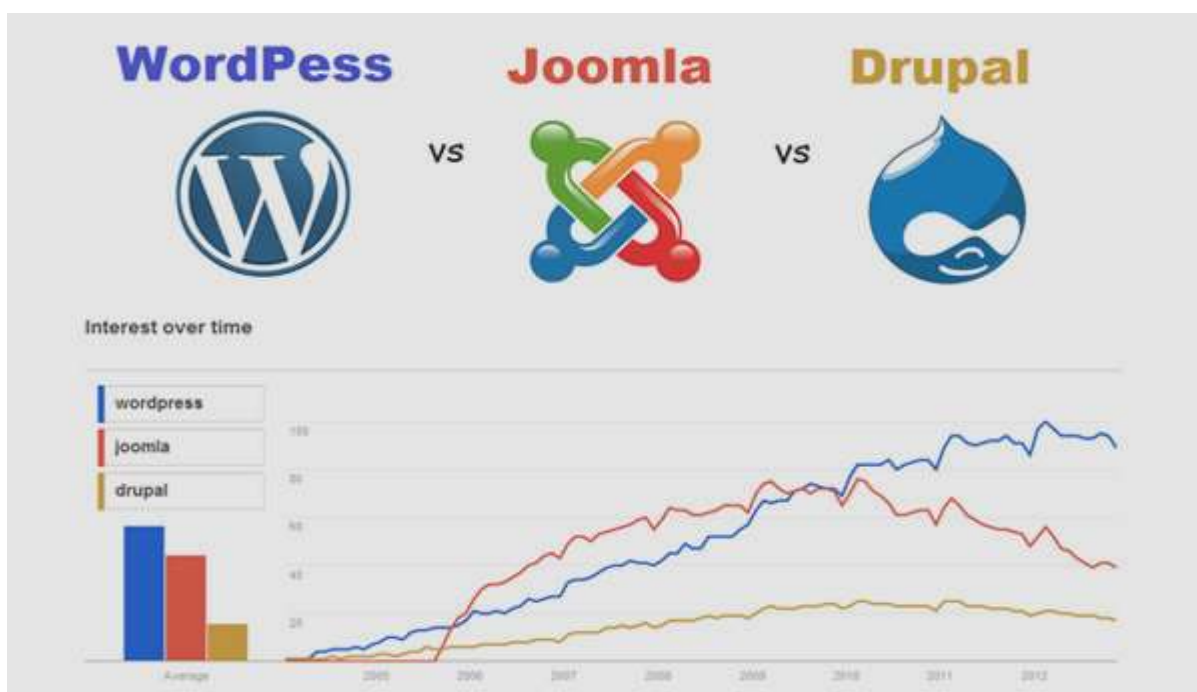
Joomla!

Joomla! е много подобна на WordPress, когато става въпрос за сигурност. Тя активно отговаря на всяка уязвимост и бързо я премахва. Въпреки това, поддържането на интернет страницата и инсталирането на ъпдейтите се извършва от потребителя. Има на разположение разширения, с които да се подсигури Joomla! сайта.

Drupal

Drupal предприема много сериозен подход към сигурността. Уязвимите места в сигурността се публикуват на сайта на Drupal, след като бъдат открити и защитени. Налице е разбирането, че Drupal сайтовете не са често обект на хакерски атаки, но това се дължи на факта, че самата CMS не е толкова популярна, колкото Joomla! или WordPress.

WordPress е най-популярната от трите разглеждани CMS поради леснотата на използване. От периода на създаване до сега WordPress е била свалена около 196 млн. пъти, Joomla! - 90 млн. пъти, а Drupal - 35 млн. (Фиг. 6)



Фиг. 6 Интерес към WordPress, Joomla! и Drupal през годините³⁸

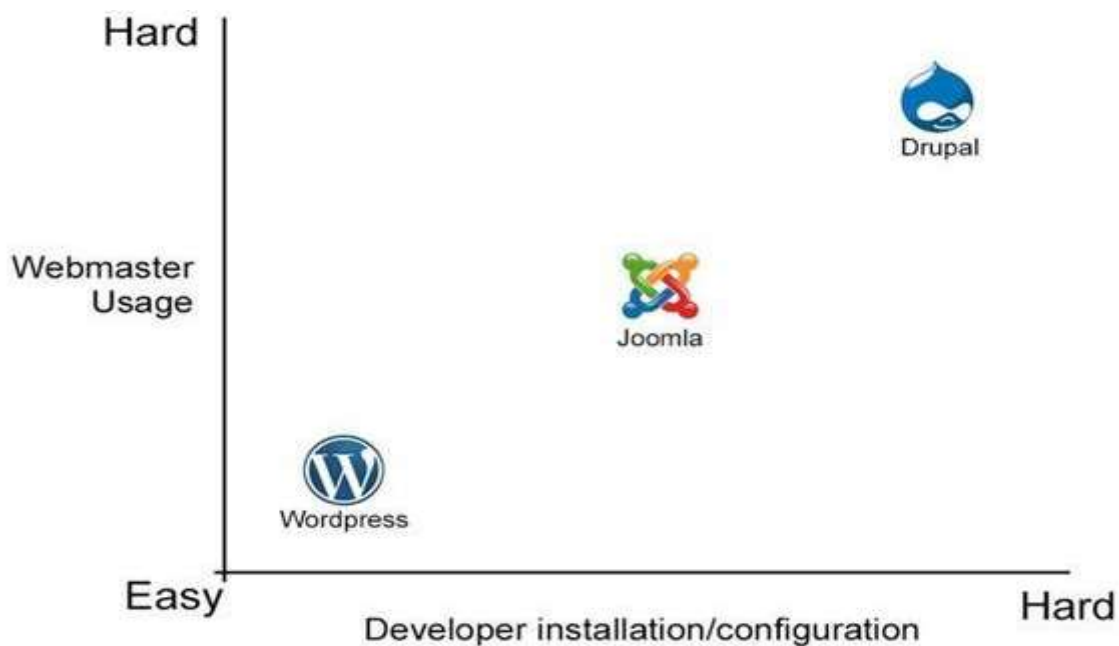
На Таблица 1 е представена обобщена информация за сравнителните характеристики на CMS приложенията – Drupal, Joomla!, WordPress.

Таблица 1. Сравнение между основните характеристики на Joomla!, Drupal, WordPress

<u>Характеристики</u>	<u>Drupal</u>	<u>Joomla!</u>	<u>WordPress</u>
<i>Сложност при използване</i>	Средна	Средна	Малка
<i>Оптимизация на търсенето</i>	ДА, изисква се добавка	ДА, изисква се плъгин	ДА
<i>Благоприятна среда за разработчиците</i>	ДА	ДА, по-ограничена от Drupal	НЕ
<i>Управление на множество сайтове с една инсталация</i>	ДА	НЕ	НЕ, изисква WordPress MU
<i>Блогове</i>	ДА	ДА, по-ограничена от WordPress	Да, отлична платформа за блогове
<i>Отделна област за администриране</i>	ДА	ДА	ДА
<i>Поддръжка от общността</i>	ДА	ДА	ДА
<i>Мащабируемост</i>	ДА	ДА	НЕ, обикновено се използва за сайтове с еднородно съдържание

³⁸ 07.03.2018 <http://www.allhostsreviews.com/web-hosting-blog/comparing-wordpress-joomla-and-drupal/>

На фигурата по-долу е показано нагледно как се разполагат трите системи според посочените основни критерии. WordPress е в единия край на диапазона като най-лесен за употреба, но неподходящ за комплексни проекти, Joomla! е в средата, и Drupal е на другия край на диапазона като най-труден за използване и най-подходящ за сложни проекти.



Фиг. 7 Сравнение на Drupal, WordPress, Joomla!³⁹

От Фиг. 7 става ясно, че не само е по-лесно да се инсталира Joomla! от Drupal, но и да се конфигурира. Също така, по-лесно е разработването на потребителска функционалност за Joomla, отколкото за Drupal и WordPress. За да се разработи висококачествен продукт по поръчка, може лесно да се промени някой от трите разглеждани CMS, за да се направи това, което се изисква, но то е за сметка на сигурността, поддръжката и надеждността на системата.

WordPress е по-подходяща за блогове, но също така предлага плъгини, които я правят подходяща и за конвенционални уебсайтове. Собствениците на уебсайтове не е нужно да притежават много умения в кодирането, за да управляват WordPress уебсайта ефективно. Качването на съдържание на WordPress е също така много по-

³⁹Ahmed, N., Comparing Joomla!, Wordpress, Drupal, Article, January 2016, 07.03.2018
<http://www.allhostsreviews.com/web-hosting-blog/comparing-wordpress-joomla-and-drupal/>

лесно в сравнение с Drupal или Joomla!. Drupal се основава изцяло на отворено кодиране и може да бъдат използвана ефективно, само от добър програмист. Joomla! стои по средата на двете крайности.

Смята се, че Drupal е по-подходящ за сложни проекти от Joomla! и WordPress. Това схващане се базира на оценката на CMS от гледна точка на уебмастъра, а не на програмиста. Drupal дава възможност за модулен подход за дизайна на потребителския интерфейс и на въвеждането на съдържание. Уеб страниците, разработени с Drupal са създадени с помощта на блокове и изгледи. Това дава гъвкавост на уебмастъра при създаването им без необходимост от програмни модули на Drupal. Joomla! предоставя подобни инструменти (компоненти и модули) за уебмастър и не е толкова трудно като при Drupal за опитен уебмастър да създаде нови сложни уеб страници. При WordPress няма толкова много опции, но това се компенсира с по-улеснената работа с нея.

WordPress и Drupal предлагат голям брой теми (съответно над 10 000 и 3000), докато Joomla! предлага само 1500. WordPress и Drupal превъзхождат Joomla! и от гледна точка на безплатните плъгини (40000 и 24000 за WordPress и Drupal; 7000 за Joomla). Докато WordPress изисква най-малко време, за да се инсталира, разликата между времето, необходимо да се инсталира Joomla! или Drupal е пренебрежително малка (приблизително 10 минути за двете).⁴⁰

В днешния свят, където всичко се управлява от софтуер, гъвкавостта при преместване на съдържание на уеб страница и наличието на модулен потребителски интерфейс не е достатъчно. Тъй като потребностите на бизнеса нарастват, функционалността на CMS трябва да бъде достатъчно гъвкава, за да отговори на тези изисквания.

Персонализирането на Drupal не е лесно, тъй като той се базира на лошо проектирана рамка, докато Joomla е базирана на добре разработена, обектно-ориентирана MVC (model-view-controller) рамка. Тя включва и няколко примерни дизайна.

⁴⁰ Usage statistics and market share of WordPress for websites, Technological Survey, 07.03.2018
<https://w3techs.com/technologies/details/cm-wordpress/all/all>

База данни

- В Drupal изгледите се съхраняват в базата данни. Това означава, че не може да има контрол на версиите и разработчиците не могат да работят заедно по създаване на дизайните.
- Всеки нов тип съдържание в Drupal отговаря на няколко таблици с бази данни. Това означава, че структурата на базата данни се променя всеки път, когато уебмастъра създава и променя типове съдържание. Това е голям проблем за разработчиците, които предпочитат да създават диаграми единица - връзка (ERD) при изграждане на уеб приложенията. Така никога не можете да се разчита на създаденото ERD, защото следващия път, когато се отвори базата от данни, броят на таблиците и схемата ще бъде различен.
- В Drupal log файловете се съхраняват в базата данни. Всички съвременни системи съхраняват log файловете си във файлове. Съхраняване на log файлове в базата данни означава, че те са много трудни за достъп, анализ и профилиране. Този процес е бавен и отнема много дисково пространство за съхранение. Това прави базата данни на системата ненужно голяма и неефективна. При висок трафик на сайта е почти невъзможно да се направи търсене и анализ на тези файлове. Освен това, не може да се поддържа log-rotation и архивиране на старите log файлове.

Дизайн

Joomla! е обектно ориентирана, но Drupal се основава на старо процедурно програмиране. Drupal използва модели на дизайн, които са стари и познати като лоша практика. Joomla! прилага модели, които са модерни, представляват добри практики и са използвани от най-добрите системи като symfony2, Zend и фирмени езици за програмиране като java:

- Обектово ориентирани (включително полиморфизъм, капсулиране, унаследяване и др.);
- MVC (Model View Control);

- Event Driven, Event Dispatcher и Observer;
- Singleton.

Някои от моделите дизайн, прилагани и в двете системи като DBAL (Database Abstraction Layer) са направени по-добре в Joomla!. На Joomla! DBAL е почти толкова добър, колкото ORM (Object-relational mapping).

Прилагането на тези модерни практики се дължи на непрекъснато усъвършенстване на рамката и CMS, което Joomla! са направили през годините, докато при Drupal не се наблюдават резки промени в структурата.

Основна архитектура

Joomla! има много чисто ядро, докато кодът на Drupal е прекалено объркан. Joomla! има йерархията на дърво. Joomla! ядрото е основното стъбло. Системата има клонове (API), на които могат да се прикачват други клонове (компоненти) или листа (модули), така че един компонент може да бъде интегриран едновременно и в системата за сигурност на Joomla! (Access Control Levels), и във функциите за администриране на статиите. Като един клон, той може да има подклонове и дори може да бъде мост към някой друг клон. Като в крайна степен има минимален контакт между различните клонове на един компонент.

В Drupal формата е до голяма степен кръгова с множество точки по повърхността, наречени модули. При изграждане на разширение, има възможност то да се включи в някой или във всички от тези модули. Тази тясна интеграция означава, че всички контактуват един с друг. Тази архитектура е по-малко елегантна и намалява качеството на кода на всички компоненти. С този вид структура поддръжка в текущ и дългосрочен план се превръща в проблем.

Изпълнение и кеширане (caching)

Drupal прави средно около 100 заявки за зареждане към базата данни на една страница. Заради тези и други особености на изпълнението, свързани с архитектурата на сайта, всичко трябва да се екстракта и кешира, което изисква допълнително хардуер и ресурси. Joomla! е много по-лека и оптимизирана. Тя има много по-бързо ядро. Препоръчваната памет при Joomla! е 512MB, докато при Drupal е 2GB.

Повечето бази за сравнение доказват, че без кеширане, Joomla! е по-бърза и изисква по-малко ресурси от Drupal. Въпреки това, някои смятат, че при кеширане Drupal е по-бърз. Но ако Joomla! кеширането е разработено и правилно конфигурирано от експерти, може да надмине Drupal, дори когато кеширането не е активно. Също така Joomla! кеширането е много по-просто и изисква по-малко ресурси, отколкото при Drupal, което прави Joomla! лесна за използване, обновяване и персонализиране.

Всички по-горе индикации показват, макар и в противоречие с общоприетото мнение, че Joomla! е много по-подходяща за разработването на по-големи и по-сложни уебсайтове, отколкото Drupal. Казаното дотук обуславя и широкия кръг от предимства, до които може да доведе внедряването и използването на една от тези системи.⁴¹

- Намаляване на разходите по поддръжка на сайта;
- Добре организиран процес по създаване на съдържанието;
- По-бързо създаване на нови страници и обновяване на страници;
- По-голяма съгласуваност;
- Подобряване на навигацията на сайта;
- По-голяма гъвкавост на сайта;
- Поддръжка на децентрализирано създаване на съдържание;
- По-голяма сигурност;
- Намаляване на дублирането на информация;
- По-големи възможности за разрастване на сайта.

По този начин всяка една организация с интернет сайт може да спести време и средства за обучение на персонала за публикуване на съдържание. Намалява се също работата на IT отдела по промени в уебсайта. Системата намалява и времето за публикуване, позволявайки по-бърз достъп до вече публикуваното съдържание.

⁴¹ Patel, S., Rathod, V., Parikh, S., Joomla. Drupal and WordPress - A Statistical Comparison of Open Source CMS, 2011, p.182, 07.03.2018 <http://dc.dyu.edu.tw/Tusi/files/Joomla,%20Drupal%20and%20WordPress%20-%20a%20statistical%20comparison%20of%20open%20source%20CMS.pdf>

Това е важен резултат за модерната организация. Колкото по-бърз е достъпът до публикуваното съдържание, толкова по-актуално и значимо е то.

Освен гореизброените, особено важно предимство, което системата за управление на съдържанието може да осигури е подпомагането на бизнес целите и стратегиите на организацията, като например повишаване на продажбите, увеличаване на потенциалните клиенти, повишаване на доверието на потребителите и подпомагане на връзките с клиенти.

Въпреки тези предимства и популярността на Drupal, Joomla! и WordPress, те не предоставят някои от общите характеристики придаващи уникалност, от които всеки уеб сайт има нужда. Това са:

Уникален дизайн и визия

Drupal и Joomla! CMS са приложения, базирани на шаблони. Те са изградени от програмисти, а не от дизайнери или хора с опит в маркетинга и дизайна. Възможно е да се създаде сайт с уникален брандинг, използвайки Drupal или Joomla!. Въпреки това, когато тези приложения са били създадени, не е имало акцент върху разработването на уникални характеристики на брендиране. При Drupal, Joomla! и WordPress този процес е трудоемък и отнема много време за изпълнение в даден сайт.

Уникални страници за отделните категории

Повечето сайтове (особено тези на предприятия и институции) изискват уникален страници за Продукт, Услуги или Категория. Въпреки че това може да се постигне с Drupal, Joomla! и WordPress, процесът ще коства много усилия и време. Тази опция не е включена в основната функционалност на нито една от разглежданите CMS.

Интуитивно и лесно администриране

Drupal и Joomla! имат много „разпръснато“ и объркващо администриране. Това е така, защото и двете са били сглобени заедно от програмисти и не е имало никакви забележими усилия за улесняване на използваемостта на тези приложения.

Все пак администрирането на Joomla! е по-добро от това на Drupal, но не е достатъчно да създаде конкурентно предимство за крайния потребител пред Drupal. WordPress е с най-добър и интуитивен дизайн на административния контролен панел.

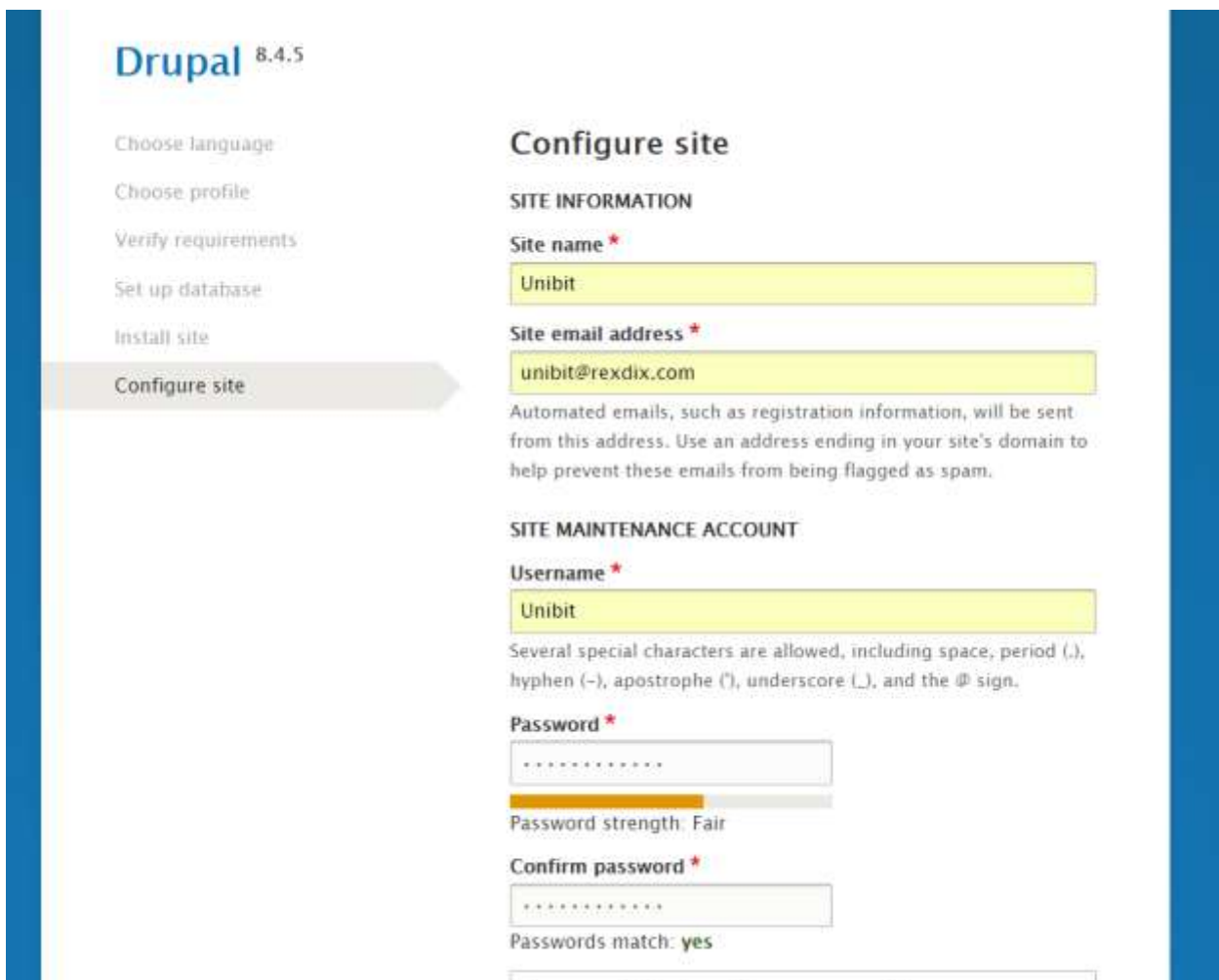
2.1 Примерни уебсайтове изградени върху Drupal, WordPress и Joomla

Drupal уебсайт - <https://drupal.rexdix.com/>

Съдържанието за всички примерни уебсайтове е взето от уебсайта на УниБИТ⁴². За използването на Drupal, Joomla и WordPress е нужно хостинг пространство, върху което да има достатъчно свободно място за да инсталирате програмата. Както и за другите две системи, са нужни MySQL сървър, PHP и уеб сървър (Apache). Drupal може да работи и с други уеб сървъри като Nginx, Microsoft IIS и Hiawatha⁴³. Самата инсталация не отнема повече от 2-5 минути. По време на инсталационния процес, програмата ще изиска да се въведе името на базата данни, както и да се създаде администраторския акаунт за системата.

⁴² <https://www.unibit.bg/>

⁴³ <https://www.drupal.org/docs/8/system-requirements/web-server>



След като Drupal е инсталирана, системата може да бъде администрирана чрез влизане с потребителското име и парола, създадени по време на инсталацията. Създаването на първоначална страница може да бъде направено през секцията “Content” -> Add content -> Basic page. След въвеждане на заглавие и съдържание на страницата, тя може да се публикува веднага или да се запази за по-късно публикуване. Тъй като, Drupal предлага възможности за изграждане на големи портали и голямо количество съдържание, той работи с тъй наречените “nodes“. За да направим страницата главна и видима при отваряне на сайта, е нужно да се избере опцията “Promoted front page”.

Home » UnibIT

Title *

UnibIT

Body (Edit summary)

Университетът по библиотекознание и информационни технологии (УниБИТ) е държавен университет, пресъобразуван на 29 IX 2010 г. с решение на Народното събрание и е приемник на Специализираното висше училище по библиотекознание и информационни технологии, гласувано от Народното събрание през 2004 г., на Държавния библиотекарски институт, създаден през 1950 г., на Популювския институт по библиотечно дело (ПИБД), на Колежа по библиотечно дело (КБД) и на Колежа по библиотекознание и информационни технологии (КБИТ).

УниБИТ обучава по [бакалавърски, магистърски и докторски програми](#) за образователните и научни степени, по които има акредитация.

УниБИТ разполага със съвременна материална база за подготовка на своите студенти. Специално проектираната и построена сграда включва тържествена зала; специализирани семинарни кабинети; аудитории; [компютърна лаборатория „Дяон Атанасов“](#); с модерни компютърни зали с локални мрежи и с постоянен достъп до Интернет, в които се изучават съвременни информационни и комуникационни технологии; Библиотечно-информационен център с читалня за 150 души, чийто специализиран библиотечен фонд е над 52 000 библиотечни единици, и книгохранилище за 80 000 документа; IP Point – информационно-консултантски център по интелектуална собственост; Карьерен център за подпомагане на професионалната реализация на студентите; Учебно-демонстрационна музейна сбирка „Духовност и лидерство“; музейна сбирка на съвременната българска духовност; издателство „За буквите – О писмента“; параклис „Св. Николай Чудотворец“; фонетичен кабинет; кабинети за преподавателите и за административния персонал; заседателна зала; модерен физкултурен комплекс със салон; тенис кортове; зала за фитнес; стол; кафе-клуб.

Published
 Last saved: 03/08/2018 - 12:33
 Author: Unibit
 Create new revision
 Revision log message
 Briefly describe the changes you have made.
 MENU SETTINGS
 URL PATH SETTINGS
 AUTHORIZING INFORMATION
PROMOTION OPTIONS
 Promoted to front page
 Sticky at top of lists

Създаването на потребители в Drupal се случва през секцията „People”. Интересна опция при създаването на потребител е възможността този потребител да има собствена форма за контакти. По този начин, посетителите могат да се свържат с автора на статията без той да обявява своя имейл адрес на страницата. Също така, се намалява значително броя на спам съобщения изпратени към имейл адреса на потребителя, ако той е видим на уебсайта.

Add user ☆

Home » Administration » People

This web page allows administrators to register new users. Users' email addresses and usernames must be unique.

Email address

unibit@rexdix.com

A valid email address. All emails from the system will be sent to this address. The email address is not made public and will only be u

Username *

unibit1

Several special characters are allowed, including space, period (.), hyphen (-), apostrophe ('), underscore (_), and the @ sign.

▼ **CONTACT SETTINGS**

Personal contact form
 Allow other users to contact you via a personal contact form

Инсталиране на нови теми (themes) в Drupal става през секцията „Appearance”. Самият процес по инсталиране и администриране на тема е

сравнително сложен. За да се инсталира нова тема е нужно да въведете директен линк или да качите собствен файл от компютъра. Линкове към теми могат да бъдат намерени на сайта на Drupal⁴⁴.

Install new theme ☆

[Home](#) » [Administration](#)

You can find [modules](#) and [themes](#) on [drupal.org](#). The following file extensions are supported: *zip tar gz bz2*.

Install from a URL

<https://ftp.drupal.org/files/projects/bootstrap-8.x-3.10.tar.gz>

For example: <https://ftp.drupal.org/files/projects/name.tar.gz>

След като новата тема е инсталирана, нейните настройки и дизайн могат да бъдат свободно променяни от потребителя.

[Home](#) » [Administration](#) » [Appearance](#) » [Appearance settings](#)

These options control the display settings for the *Bootstrap* theme. When your site is displayed using this theme, these settings will be used.

THEME UPDATES

Bootstrap Settings

General
Responsive Images

Components
Breadcrumbs, Navbar: Normal

JavaScript
Modals (Bridged), Popovers, Tooltips

Advanced
CDN provider: *jsDelivr*; 3.3.7

▼ BUTTONS

- Colorize Buttons**
Adds classes to buttons based on their text value.
See also: [Buttons](#); [hook_bootstrap_colorize_text_alter\(\)](#)
- Iconize Buttons**
Adds icons to buttons based on the text value
See also: [hook_bootstrap_iconize_text_alter\(\)](#)

Default button size
Normal ▼
Defines the Bootstrap Buttons specific size

▼ CONTAINER

- Fluid container**
Uses the `.container-fluid` class instead of `.container`.
See also: [Fluid container](#)

⁴⁴ https://www.drupal.org/project/project_theme

Joomla уебсайт - <https://joomla.rexdix.com/>

Инсталирането на Joomla отнема почти същото време, което е нужно и за Drupal. Инсталационният процес е подобен, като Joomla предоставя няколко допълнителни опции като:

- Премахване на инсталационната папка след завършване на процеса по инсталиране с цел подобряване на сигурността;
- Показване на различни сървърни модули и разширения;
- Блокиране на публичния достъп за да не се влоши индексирването на сайта в процеса на разработка и тестване;
- Допълнително инсталиране на други езици.

Pre-Installation Check

If any of these items are not supported (marked as **No**) then please take actions to correct them.
You can't install Joomla! until your setup meets the requirements below.

PHP Version >= 5.3.10	Yes
Magic Quotes GPC Off	Yes
Register Globals Off	Yes
Zlib Compression Support	Yes
XML Support	Yes
Database Support: (mysql, sqlite, pdomysql, pdo)	Yes
MB Language is Default	Yes
MB String Overload Off	Yes
INI Parser Support	Yes
JSON Support	Yes
configuration.php Writeable	Yes

Recommended settings:

These settings are recommended for PHP in order to ensure full compatibility with Joomla.
However, Joomla! will still operate if your settings do not quite match the recommended configuration.

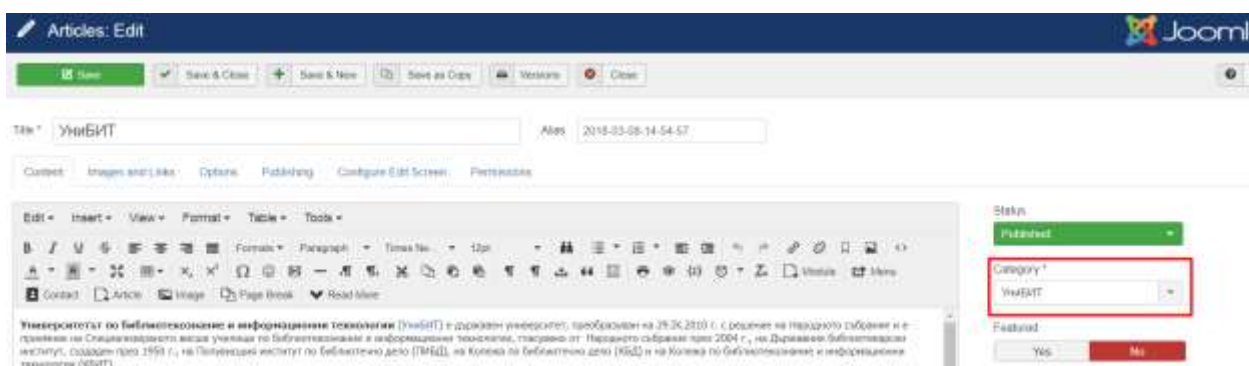
Directive	Recommended	Actual
Safe Mode	Off	Off
Display Errors	Off	On
File Uploads	On	On
Magic Quotes Runtime	Off	Off
Output Buffering	Off	Off
Session Auto Start	Off	Off
Native ZIP support	On	On

Административният панел на Joomla е по-подреден и по-лесен за навигация в сравнение на този в Drupal. Бутоните, секциите и менютата са логично подредени. Joomla предлага различни варианти за контрол на достъпа⁴⁵. Могат да се създават нови потребители с различни нива на достъп, което предлага високо ниво на сигурност и малък шанс от злонамерени действия.

⁴⁵ https://docs.joomla.org/J3.x:Access_Control_List_Tutorial

Level Name	User Groups Having Viewing Access	ID
Public	Public	1
Guest	Guest	5
Registered	Manager, Registered, Super Users	2
Special	Author, Manager, Super Users	3
Super Users	Super Users	6

Също както Drupal, Joomla е удобна платформа за администриране на големи уебсайтове. Joomla използва т.нар категории за контролиране на съдържанието. Тези категории могат да се нарекат още папки, които са предимно използвани от самата система за подреждане на различните страници, статии и други елементи на уебсайта. Всяка страница на уебсайта, трябва да бъде сложена в категория за да бъде видима. Нова страница (статия) може да бъде създадена през менюто Content -> Articles -> New⁴⁶. Има голям избор от настройки, които могат да бъдат променяни за всяка статия, но най-важната, която трябва да бъде избрана за да бъде видима страницата е статията да бъде сложена в категория през падащото меню Category:.



След като бъде инсталирана Joomla, категорията, която е създадена по подразбиране е Uncategorized. Нови категории могат да бъдат създавани през секцията Content -> Categories -> Add new category. Също както за нови страници, тук са налични много варианти за категории, които могат да бъдат променяни и след като категорията е създадена. Менютата и наличните опции в административния панел са по-добре подредени за уебмастъра, което улеснява поддържането на системата без да е нужен специалист. Документацията на Joomla е

⁴⁶ https://docs.joomla.org/Adding_a_new_article

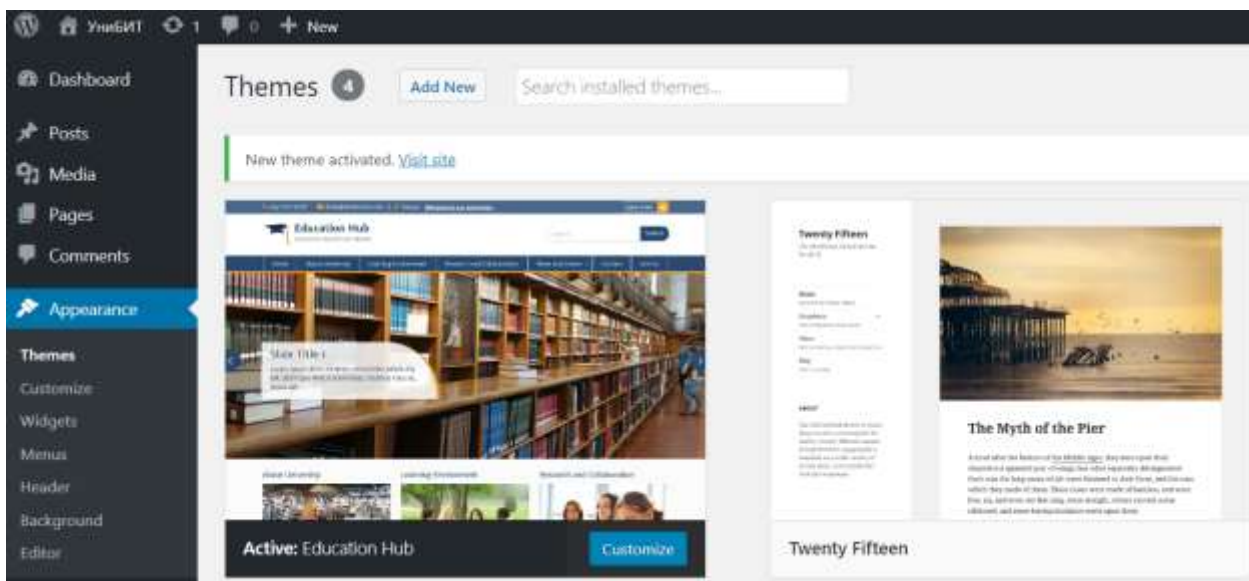
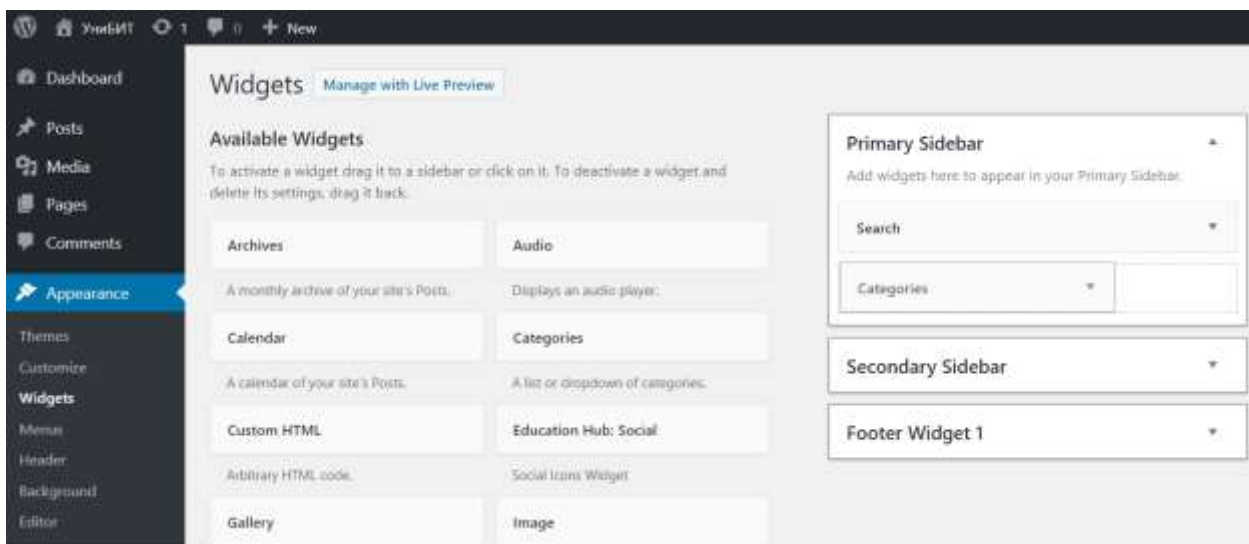
подредена и са описани всички опции и възможности на системата, която я прави удобна и лесна за научаване от потребителите.

WordPress уебсайт - <https://wordpress.rexdix.com/>

WordPress е най-лесната за инсталиране система в сравнение с Joomla и Drupal. Изискванията са отново същите като на другите две системи (PHP, MySQL и Apache уеб сървър). Също като при другите системи, при инсталационния процес се задава потребителско име и парола за администратора на уебсайта. WordPress има най-благоприятна среда за администриране и това я прави най-използваната система за персонален уебсайт. Все пак, има и големи уебсайтове базирани върху WordPress, като например сайта на компанията Уолт Дисни (<https://thewaltdisneycompany.com/>), Sony Music (<https://www.sonymusic.com/>) и други. WordPress има огромен избор от теми (themes) и добавки (плъгини). От голяма важност е всички добавки да бъдат проверени преди да бъдат инсталирани. Самото ядро на WordPress е много добре конструирано, но добавките не винаги са съвместими с последната версия на системата и могат да предизвикат проблеми в темата или функциите на уебсайта. WordPress предлага отлично ниво на сигурност на потребителите. Важно е администраторът на уебсайта да поддържа всички темплейти и допълнителни приложения обновени до техните последни версии за да избегне проблеми със сигурността. За съжаление, много от приложенията са написани от неопитни програмисти и съдържат много уязвими места, които са лесна атака за хакерите. За създаване на съдържание, WordPress предлага възможност за публикуване на отделни страници и статии на всяка от страниците:



Един от недостатъците на WordPress е, че е базиран на теми, които могат да бъдат използвани и на други сайтове, което от своя страна губи уникалността на даден уебсайт. Съществува голям брой безплатни и платени теми, които могат да бъдат персонализирани до някаква степен, като избор на менюта, background картинки и други настройки.



Заклучение

В магистърската теза е анализирана същността на системите и процеса на управление на съдържанието, направено е сравнение на трите най-използвани CMS с отворен код – Drupal, WordPress и Joomla! по определени основни критерии и са разгледани подробно техните предимства и недостатъци.

На база на този анализ могат да се направят следните изводи:

1. Управлението на съдържанието е основна част от всяка организация, като един CMS продукт може да покрие множество категории на съдържанието;
2. CMS с отворен код се фокусират върху приложните функции, а не върху интеграцията;
3. Drupal е подходяща за информативни и интранет уебсайтове, тъй като изисква по-малко ресурси; Drupal предполага по-голяма гъвкавост, но администрацията е доста сложна за уебмастъра. Клиентите трябва да наемат експерт в Drupal и общите потребители на сайта не могат лесно да бъдат обучени да използват Drupal, както могат да бъдат обучени, за да използват Joomla!.
4. За интранет сайтове, с множество обекти се препоръчва използването на Joomla!, понеже тя се справя с натоварването по-добре от Drupal, особено след кеширане; Joomla! има много по-голяма общност от разработчици в сравнение с Drupal.
5. WordPress се оказва най-универсалната CMS от трите разглеждани в разработката, най-вече при сайтове, които са създадени основно с цел да показват информация, такива, които трябва да бъдат гъвкави, за да отговарят на променящите се в реално време нужди на потребителите или сайтове за електронна търговия.

6. За начинаещи, WordPress е определено най-добрият вариант, тъй като предлага достъпен интерфейс и дори потребители без технически познания могат да обновяват съдържанието на WordPress уебсайта. Освен това системата има богата колекция от теми и други функции, които са лесни за използване от обикновените потребители. Joomla! е особено популярна сред потребителите на социалните мрежи, тъй като тя разполага с платформа и функции, които могат да се използват ефективно за създаване на онлайн общество и улесняване на взаимодействието. Drupal се използва най-вече от тези, които се занимават с бек-офис задачи (счетоводство, човешки ресурси и т.н.). Маркирането става изключително лесно и организирането на съдържанието е по-ефективно и точно.

Използвана литература

1. Ahmed, N., Comparing Joomla!, WordPress, Drupal, Article, January 2016, 07.03.2018 <http://www.allhostsreviews.com/web-hosting-blog/comparing-wordpress-joomla-and-drupal/>
2. Boiko, B., Understanding Content Management, Metatorial Services Inc. & HungryMinds Inc, 2012, p.43, 07.03.2018
http://www.metatorial.com/downloads/Boiko_Wp_UnderstandingContentManagement.pdf
3. Classen, H., FUNDAMENTALS OF SOFTWARE LICENSING, IDEA: The Journal of Law and Technology 1996, p.355, 12.02.2018
<http://euro.ecom.cmu.edu/program/law/08-732/Transactions/Fundamentals.pdf>
4. Franziska Frey, Howard Vogl, Digital Asset Management— A Closer Look at the Literature, Printing Industry Center NY, March, 2005, p.5, 12.02.2018
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.369.7294&rep=rep1&type=pdf>
5. Geisert, S., Railsback, S., Content Management System User Guide, COLORADO STATE UNIVERSITY, 2008, pp.31-35, 12.02.2018
http://www.code.colostate.edu/Data/Sites/5/UserFiles/MojoPortal/docs/CMS_Manual_v2.pdf
6. Halverson, K., Rach, M., Content Strategy, New Riders, 2nd edition, 2012, p.41, 07.03.2018
<http://ptgmedia.pearsoncmg.com/images/9780321808301/samplepages/0321808304.pdf>
7. Honeyager, J., History of Joomla, Article, 07.03.2018
<http://expertwebprofessionals.com/news-data-blogs-web-experts/articles-about-joomla/joomla-background/103-history-of-joomla.html>
8. Luinenburg, L., Jansen, S., Designing Web Content Management Systems Using the Method Association Approach, Department of Information and Computing Sciences, Utrecht University, The Netherlands, 2008, p.16, 07.03.2018 <http://ceur-ws.org/Vol-389/paper08.pdf>

9. McKeown, S., Wordpress – Freedom, Community and the Business of Open Source, p.2, 12.02.2018 <https://wordpress.org/about/history/chapter3.pdf>
10. Merker, L., A Guide to Web Content Management System Evaluation, White Paper, pp.3-7, 12.02.2018
https://omniupdate.com/resources/pdfs/white_papers/cms_evaluation_guide.pdf
11. Mican, D., Tomai, N., Coroş, R., Web Content Management Systems, a Collaborative Environment in the Information Society, Informatica Economică vol.13, no 2/2009, p. 78, 07.03.2018 <http://revistaie.ase.ro/content/50/003%20-%20mican.pdf>
12. Michelinakis, D., Open Source Content Management Systems: An Argumentative Approach, The University of Warwick Warwick Manufacturing Group, 2004, pp.209-211, 11.01.2018 <http://www.henworx.de/pdf/oscms-report.pdf>
13. Patel, S., Rathod, V., Parikh, S., Joomla. Drupal and WordPress - A Statistical, Comparison of Open Source CMS, 2011, p.182, 07.03.2018
<http://dc.dyu.edu.tw/Tusi/files/Joomla,%20Drupal%20and%20WordPress%20-%20a%20statistical%20comparison%20of%20open%20source%20CMS.pdf>
14. Pico, P., Silva, A., WORKFLOWS IN CONTENT MANAGEMENT SYSTEMS, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 2008, pp.122-126, 07.03.2018 <http://isg.inesc-id.pt/alb/static/papers/2008/i71-pdsp-webist2008.pdf>
15. Pico, P., Silva, A., WORKFLOWS IN CONTENT MANAGEMENT SYSTEMS, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 2008, pp.122-126, 07.03.2018 <http://isg.inesc-id.pt/alb/static/papers/2008/i71-pdsp-webist2008.pdf>
16. Shafie, H., DITA for the Impatient, Pixware SARL, April 8, 2016, p.18, 01.03.2018 <http://www.xmlmind.com/tutorials/DITA/DITA.pdf>
17. About Metadata, White Paper, 09/2004 p. 1, 12.02.2018
http://www.adobe.com/digitalimag/pdfs/about_metadata.pdf
18. CMS Workbook, MichiganTech, April 2016, p.6, 12.02.2018
http://www.mtu.edu/mtu_resources/cms-workbook/cms-workbook.pdf

19. Enterprise Content Management (ECM) Overview, A White Paper, Intergraph, 07/2011, p.9, 12.02.2018
<http://www.intergraph.com/assets/plugins/ppmcollaterals/files/EMCOOverview.pdf>
20. How to evaluate a content management system, 12.02.2018
http://www.steptwo.com.au/papers/kmc_evaluate/
21. Managing and maintaining a CMS website, University of Bristol, Reference Guide, 2010, p. 8, 07.03.2018
<http://www.bristol.ac.uk/is/media/training/documentation/webcms-1/webcms-1r.pdf>
22. Usage statistics and market share of WordPress for websites, Technological Survey, 07.03.2018 <https://w3techs.com/technologies/details/cm-wordpress/all/all>
23. 12.02.2018 <https://assoc.drupal.org/about/history>
24. 12.02.2018 <https://codex.wordpress.org/History>
25. 12.02.2018 <https://websiteworker.com/drupal-history-stats-and-facts>
26. 12.02.2018 <https://www.drupal.org/about/history>
27. 12.02.2018 <https://www.drupal.org/about/mission-and-principles>
28. 12.02.2018 <https://www.drupal.org/node/114846>
29. 01.03.2018 <https://www.joomla.org/announcements/release-news/5654-joomla-3-5-is-here.html>
30. <https://www.unibit.bg/>
31. <https://www.drupal.org/docs/8/system-requirements/web-server>
32. https://www.drupal.org/project/project_theme
33. https://docs.joomla.org/J3.x:Access_Control_List_Tutorial
34. https://docs.joomla.org/Adding_a_new_article