



**УНИВЕРСИТЕТ ПО БИБЛИОТЕКОЗНАНИЕ И  
ИНФОРМАЦИОННИ ТЕХНОЛОГИИ**

**КАТЕДРА „ИНФОРМАЦИОННИ СИСТЕМИ И ТЕХНОЛОГИИ”  
СПЕЦИАЛНОСТ „ИНФОРМАЦИОННИ ТЕХНОЛОГИИ В СЪДЕБНАТА  
АДМИНИСТРАЦИЯ“**

**ДИПЛОМНА РАБОТА**

**на тема:**

**Изграждане на динамичен уеб сайт за детска градина №182 „Пчелица”**

**Дипломант:**

Жанета Кирилова

редовно обучение

Ф.№ 34-итср

**Научен ръководител:** .....

(гл.ас д-р Катя Рашева-Йорданова)

**София**

**2016**



## УНИВЕРСИТЕТ ПО БИБЛИОТЕКОЗНАНИЕ И ИНФОРМАЦИОННИ ТЕХНОЛОГИИ

### ДЕКЛАРАЦИЯ

От Жанета Кирилова

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

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

Дата:

Подпис .....

(Жанета Кирилова)

## РЕЗЮМЕ

Кирилова, Ж. Изграждане на динамичен уеб сайт на ЦДГ №182 „Пчелица”. Научен ръководител гл. ас д-р К. Рашева-Йорданова. София. 2016. Катедра „Информационни системи и технологии”. Специалност „Информационни технологии в съдебната администрация”. УНИБИТ. 94стр.

Брой източници – 25, брой фигури - 29, брой таблици – 15.

**Целта** на дипломната работа е да бъде разработен уеб сайт на детска градина, базиран на система за управление на съдържанието.

**Ключови думи:** уебсайт, уебдизайн, информационна архитектура, Joomla, CMS, контент мениджмънт система.

Настоящата дипломна работа е организирана в три глави, увод, заключение и списък на използвани източници.

В първа глава са разгледани видовете уеб сайт, разгледан е накратко процесът по изработка на уеб сайт. Представени са основните техники, които се използват за създаването му, като най-често използваните са разгледани по-подробно. На края на главата е направен избор на техника за създаване на уеб сайт отговаряща на целите и задачите за настоящата дипломна работа.

Във втора глава обстойно са разгледани видовете системи за управление на съдържанието, техните предимства и недостатъци. Направен е сравнителен анализ на трите най-популярни безплатни WCMS (Wordpress, Joomla, Drupal) и една система от по-нисък клас (Plone) и е избрана най – подходящата за изграждането на сайт за детска градина.

В трета глава е направен анализ на съществуващи сайтове на други детски градини, изведени са изискванията произтичащи след направения анализ. Също така са дефинирани специалните изисквания поставени от администрацията на детското заведение за изграждането на сайта. Накрая е представена реализацията на изграждането на динамичен уеб сайт с помощта на системата за управление на съдържанието Joomla!.

# Съдържание

РЕЗЮМЕ .....	3
УВОД	7
ГЛАВА ПЪРВА: АНАЛИЗ НА ТЕХНИКИТЕ И ТЕХНОЛОГИИТЕ ЗА СЪЗДАВАНЕ НА УЕБ САЙТ.....	9
1.1. Основни характеристики на уеб сайта.....	9
1.2. Видове уеб сайтове .....	9
1.2.1. Видове сайтове според предназначението им .....	10
1.2.2. Видове уеб сайтове според приложената технология .....	12
1.3. Процес по изработка на уеб сайт.....	13
1.3.1. Уеб дизайн .....	14
1.3.2. Уеб програмиране (уеб разработване).....	16
1.4. Техники за създаване на уеб сайта.....	16
1.4.1. Езици за програмиране.....	16
1.4.1.1. HTML .....	17
1.4.1.2. DHTML .....	18
1.4.1.3. XML.....	19
1.4.1.4. Visual Basic .....	19
1.4.1.5. Visual Basic Script .....	20
1.4.1.6. JavaScript.....	20
1.4.1.7. Java .....	21
1.4.1.8. PHP .....	22
1.4.1.9. C.....	23
1.4.1.10. C++ .....	23
1.4.1.11. C#.....	24
1.4.1.12. Python .....	24
1.4.1.13. Ruby.....	25
1.4.1.14. Objective-C.....	25
1.4.1.15. ActionScript.....	25
1.4.1.16. Perl .....	26
1.4.1.17. CSS .....	27
1.4.1.18. ASP .....	27

1.4.1.19. Ajax.....	28
1.4.2. Системи за управление на съдържанието .....	29
РЕЗУЛТАТИ И ИЗВОДИ ОТ ПЪРВА ГЛАВА .....	33
ГЛАВА ВТОРА: СПЕЦИФИКИ НА КМС ОТ ГЛЕДНА ТОЧКА НА ВЪЗМОЖНОСТТА ЗА ИЗГРАЖДАНЕ НА САЙТ ЗА ДЕТСКА ГРАДИНА	34
2.1. Структура на КМС .....	34
2.2. Видове системи за управление на съдържанието	35
2.2.1. Компонент контент мениджмънт системи.....	37
2.2.2. Система за управление на образователен контент.....	38
2.2.3. Система за управление на корпоративно съдържание.....	40
2.2.4. Система за управление на документи.....	42
2.2.5. Система за управление на уеб съдържание.....	44
2.2.5.1. Видове WCMS.....	45
2.3. Комерсиални, Open Source и потребителски КМС	46
2.3.1. Комерсиални КМС.....	46
2.3.1.1. Предимства на комерсиалните КМС .....	47
2.3.1.2. Недостатъци на търговските КМС.....	47
2.3.2. Open Source CMS .....	48
2.3.2.1. Предимства на КМС с отворен код .....	48
2.3.2.2. Недостатъци на КМС с отворен код .....	49
2.3.3. Потребителски КМС.....	50
2.4. Избор на Open Source WCMS за нуждите на детска градина	50
2.4.1. CMS WordPress.....	51
2.4.2. CMS Joomla.....	54
2.4.3. CMS Drupal .....	56
2.4.4. CMS Plone .....	58
2.4.5. Сравнителен анализ .....	61
РЕЗУЛТАТИ И ИЗВОДИ ОТ ВТОРА ГЛАВА .....	64
ГЛАВА ТРЕТА: ИЗГРАЖДАНЕ НА УЕБ САЙТ ЗА ДЕТСКА ГРАДИНА №182 „ПЧЕЛИЦА“ .....	65
3.1. Дефиниране на изискванията към сайта и аспектите на информационната му архитектура .....	65

3.1.1. Изграждане на списък с изисквания към уебсайта, базиран на анализ на конкурентни сайтове.....	66
3.1.1.1. Анализ на дизайна на сайтовете.....	66
3.1.1.2. Анализ на функционалностите в сайтовете .....	70
3.1.1.3. Анализ на информацията предоставена в сайтовете .....	71
3.1.2. Изисквания към уеб сайта от страна на детската градина.....	71
3.2. Етапи от изработката на уебсайта .....	72
3.2.1. Избор и инсталация на темплейт .....	72
3.2.2. Избор и изграждане на навигационно меню.....	75
3.2.3. Създаване на начална страница и вътрешни страници.....	79
3.3. Добавяне на основните функционалности в сайта	84
3.3.1. Добавяне на карта „Карта”.....	84
3.3.2. Създаване на „Галерия” .....	85
3.3.3. Създаване на регистрационна форма и задаване на права на потребителите от администрацията на детското заведение .....	86
3.4. Публикуване и поддръжка на сайта .....	87
3.4.1. Публикуване на сайта.....	87
3.4.2. Поддържане и актуализиране .....	90
РЕЗУЛТАТИ И ИЗВОДИ ОТ ТРЕТА ГЛАВА.....	91
ЗАКЛЮЧЕНИЕ .....	92
ИЗТОЧНИЦИ.....	93

## УВОД

В днешното забързано ежедневие Интернет е едно от основните средства за информиране. Повечето хора предпочитат да проверяват за всичко в уеб пространството, за да спестят време от нуждата да посетят дадено място. Поради тази причина липсата на уеб сайт изолира фирми и организации от много потребители. Там информацията е лесно достъпна и не е ограничена от разстояние и работно време, тя е може да бъде достигната по всяко време на денонощието и от всяко място.

**Актуалността** на настоящата дипломна работа произлиза от факта, че Интернет е най-голямата медия в света. Тя е достъпна за почти всички хора и чрез нея лесно може да бъде популяризирана всяка фирма или организация. Именно от такава популяризация се нуждаят детски градини и училища застрашени от затваряне поради липса на достатъчно записани деца.

**Целта** на настоящата дипломна работа е да се изгради уеб сайт на детска градина, като се използва системата за управление на съдържание Joomla!. По този начин ще бъде предоставена информация онлайн на родители на вече записани деца и същевременно ще бъде популяризирана пред множество други хора, което може да допринесе до разрастването на детското заведение от към записани деца.

За реализиране на целта са поставени следните **задачи**:

- Представяне на основните технологии, чрез които може да бъде изграден уеб сайт и въз основа на направен анализ да бъде подбрана най-подходящата за реализацията на проекта такава
- Анализ на съществуващи уеб сайтове на детски градини въз основа на който да бъде изготвен списък с необходими функционалности и подходи при изграждане на информационната архитектура, така че разработвания проект да отговори на очакванията на потребителите.

- Изграждане на уеб сайт за детското заведение, базиран на предварително избраната технология и отговарящ на всяко от формулираните изисквания.

**Обектът** на настоящата разработка е Целодневна Детска Градина № 182 „Пчелица”.

За подготовката и изграждането на дипломната работа са използвани общо източника 25 източника – сред които книги, ръководства и онлайн статии.



# ГЛАВА ПЪРВА: АНАЛИЗ НА ТЕХНИКИТЕ И ТЕХНОЛОГИИТЕ ЗА СЪЗДАВАНЕ НА УЕБ САЙТ

За да бъде създаден един уеб сайт първо трябва да се обясни какво всъщност представлява той, какви видове има и с какви техники може да бъде създаден. В тази глава ще бъдат разгледани тези аспекти на уеб сайтовете и след анализ ще бъде подбрана най-добрата техника за изграждане на сайт за детска градина.

## 1.1. Основни характеристики на уеб сайта

Един интернет сайт представлява съвкупност от web страници. От своя страна Web страниците представляват файлове, които са написани на специален език за програмиране (например HTML, PHP, Perl и др.). Всяка страница може да съдържа текст, изображения, видео и др. Всяка страница има адрес, наричан често просто URL. Началната част на адреса е името на сайта (домейн), а останалата част от адреса показва коя е конкретната страница от този сайт.

За да се разглеждат сайтове се използват програми, които се наричат браузери. Такива програми са Internet Explorer, Mozilla Firefox, Google Chrome и др.

Основни изисквания при създаването на уеб сайт са той да бъде атрактивен, полезен и достъпен. Атрактивността ще служи да привлече потребителите. Ползата ще го накара да се върне отново, а достъпността ще му предостави необходимото удобство при използване на сайта.

## 1.2. Видове уеб сайтове

Видовете уеб сайт могат да бъдат, диференцирани според критериите:

- Предназначение
- Приложена технология

В настоящия раздел ще бъдат разгледани спецификите на всяко от изброените деления.

### 1.2.1. Видове сайтове според предназначението им

Според своето предназначение уеб сайтовете могат да бъдат поделени на следните различни категории, основните от които са посочени по-долу : [2]

- Портали;
- Информационни;
- Новинарски;
- Корпоративни;
- Сайтове за услуги;
- Електронни магазини;
- Институционални;
- Развлекателни сайтове;
- Лични сайтове;
- Други.

*Порталът* съдържа голям брой връзки, класифицирани по определен принцип и служи за отправна точка към различни други сайтове в Интернет. Тези сайтове предлагат достъп до тясно специализирана информация и линкове към сайтове със специфично съдържание. Те предоставят възможност за търсене, регистрация на потребител или създаване на електронна поща. Най-важното при порталните сайтове е тяхната функционалност. Възможността за изключително впечатляващ дизайн липсва, тъй като целта на портала е да дава бързо и лесно информация. [2]

Основна цел на *информационните сайтове* е да предоставят достъп до информация. Информационните сайтове също съдържат голям брой връзки, но графичните елементи са повече от тези при портала. Характерното за информационните сайтове е че те се различават по отношение предоставяната от тях функционалност и визия когато са насочени към различни видове аудитория. [2]

*Новинарските сайтове* са разновидност на информационните, като се различават от тях както по съдържашата се в тях информация, така и по динамиката при нейното обновяване. Информацията, представена в тях е

организирана в рубрики, които се поддържат и се считат за постоянни. В тази категория уеб сайтове е предвидена възможност за създаване на архив от публикуваните новини. [2]

*Корпоративните сайтове* предлагат голямо разнообразие на визуални и функционални решения. Те имат своя определена аудитория, както и структура на представяното съдържание. При проектирането и изграждането на тази категория уеб сайтове се използва по-креативен подход, включващ по-сложни цветови решения и разчупени дизайнерски композиции. Преобладават графични менюта и се използват графични и анимирани (flash) елементи. Корпоративният уеб сайт цели да впечатли, да информира, да заинтригува посетителите, което ще подпомогне присъединяването им като клиенти или партньори. [2]

*Сайтовете за услуги и електронните магазини* са предназначени за масово обслужване на голяма аудитория. Предлагат извършване на услуги или продажби на стоки в зависимост от категорията, в която попадат. Тези сайтове предоставят възможности за осъществяване на онлайн разплащания и онлайн поръчка. В тази категория уеб сайтове се наблюдава засилено използване на рекламния ефект. [2]

*Институционалните сайтове* имат представителни функции, тъй като се изграждат от държавни и други ведомства. Информацията, която се предоставя от тях често има нормативен характер и юридическа стойност. Целта на този тип сайтове е да информират гражданите за извършваната от институциите, които представят публична дейност. Традиционното оформление на тези сайтове е в две колони, с малко на брой цветове, направено така, че да се зарежда от възможно най-разнообразни браузъри, като се отделя особено внимание на бързото зареждане на отделните страници. [2]

При *арт сайтовете* се наблюдава голяма свобода при използването на графичните елементи, текста, навигацията както и цялостната структура на сайта и неговият интерфейс. При тези сайтове се използват авангардни цветови и композиционни решения и често компютърните екрани се превръщат в частна виртуална галерия за изкуство представено в интернет. [2]

През последните години *развлекателните сайтове* придобиха изключително голяма популярност, както сред по-младите потребители на интернет, така и на по-възрастната част от тях. Те предоставят привлекателен и интерактивен дизайн с възможности за търсене на информация и файлове. [2]

По отношение проектирането и изграждането на *личен уеб сайт* не съществуват ограничения. Тук създателите могат да правят всичко, което сметнат за подходящо стига така да задоволяват целите си. Днес е особено популярно (дори за някои професии е задължително) човек да притежава лично портфолио, публикувано в уеб. [2]

Познаването на отделните категории уеб сайт, както и основните изисквания към изграждането им с оглед тяхната функционалност, съдържание и дизайн е важен момент от проектирането на уеб сайта.

Правилният избор и използване на тези елементи предопределя изграждането на сайтове, които не само изглеждат добре, но и функционират добре. При проектирането на сайта е добре да се направи маркетингово проучване, анализиращо лидерите в областта въз основа на което да се извлекат както добрите, така и лошите практики по отношение на включеното съдържание, функционалности, оформление, дизайн и пр. Подобен подход предполага избягване на грешки допуснати от други сайтове и създава възможност да се реализира добро решение. [2]

### **1.2.2. Видове уеб сайтове според приложената технология**

Делението на уеб сайтовете според приложената технология включва:

- *Статични уеб сайтове*
- *Динамични уеб сайтове.*

При *статичния сайт* всяка страница от сайта се изгражда сама за себе си. Тя представлява самостоятелен HTML файл, в който се съдържат всички елементи от страницата - както дизайнът, така съдържанието на самата страница. Когато посетител кликне върху адреса на тази страница, сървърът директно я изпраща за разглеждане към браузъра на посетителя. Статичните уеб сайтове са най-простият вид на уеб сайт и са подходящи за малки сайтове,

които не се променят много и не изискват персонализация. Специфичното за тези сайтове е, че отсъства така наречената „администрация”, от където собственикът на сайта може да променя съдържанието му. Изборът на статичен сайт ограничава собственика му по отношение актуализацията на информацията в него, тъй като за промяната дори на една буква, е необходима намеса на дизайнер или програмист. Това само по себе си не е рентабилно за клиента, защото генерира постоянно допълнителни разходи. Ето защо в случай, че е необходимо в уеб сайта да бъдат нанасяни чести актуализации както на съдържанието, така и на функционалностите и визуалната концепция за предпочитане би било използването на динамична среда. [2]

*Динамичният уеб* сайт разполага със система за управление на съдържанието (Content Management System), която позволява хора без специфични технически познания, да променят в реално време динамичното съдържание на уеб сайта. Така добавянето на всякакъв тип съдържание се осъществява почти интуитивно от потребителите на системата, като с няколко клика могат да бъдат добавени текст, снимка, хипервръзка, таблица. За разлика от статичната страница, която съществува и е на разположение по всяко време, динамичната страница в същност не съществува постоянно, а се създава (генерира) динамично само при поискване от потребител. [2]

### **1.3. Процес по изработка на уеб сайт**

Процесът по изработка на уеб сайт включва два под-процеса – дизайн и програмиране. Най-общо казано, терминът уеб дизайн се отнася до естетическата и функционалната страна на един уеб сайт. Уеб разработчиците (програмистите), от друга страна, вземат дизайна на уеб сайта и го правят функциониращ. За целта, те използват различни програмни езици, като HTML, CSS, JavaScript, PHP и много други. Доста често обаче в днешно време уеб дизайнерите и уеб разработчиците са един и същ човек изпълняващ и двата под-процеса. [6]

### 1.3.1. Уеб дизайн

Уеб дизайнерите, задължително, трябва да започнат работата от установяването и разбирането на нуждите на клиента. След като разберат какви са желанията на клиента, те трябва да преминат към създаване на информационната архитектура на уеб сайта.

Информационната архитектура на един уеб сайт се състои в процеса на прецизно структуриране на информацията, нейното организиране в навигационни схеми, и представянето и чрез интуитивни визуални елементи. Информационната архитектура формира ядрото на един сайт, около което се изграждат всички останали негови компоненти: общата визия на сайта, функционалностите му, потребителският интерфейс. Крайната цел на информационната архитектура в един уеб сайт е да направи информацията лесно достъпна, разбираема и управляема.[13]

Всъщност, уеб дизайнът на интернет страници най-силно се приближава до архитектурата. Както основната цел на архитектурата е да проектира сграда, пълноценно изпълняваща функциите си (живеене, производство, офис нужди и т.н.), така и при уеб дизайн-а, основната цел е да се проектират уеб сайтове, които да изпълняват своите функции - да информират, да рекламират и да продават.

Като обобщение на изложеното дотук уеб дизайн на уеб сайт това е изработка на онлайн визия, съчетана с фирмена идентичност и посетителска аудитория. Уеб сайт дизайн е вид аранжиране на линкове /уеб връзки/, съдържание на уеб сайт, графика, анимация, звук и т.н. Уеб дизайн е смесица от творчески подход и техническа изработка на един сайт. Така уеб дизайн-а трябва да бъде съчетан не само с доброто представяне на информацията на уеб сайт, а и с дизайн съобразен с фирмената идентичност - цвят, лого и т.н.

Показателите за добър уеб дизайн са:[14]

- Визия на уеб дизайн: При изработка на уеб сайт важно място заема изработването на графичната визия на уеб сайта. Визия на уеб сайт - това е външният вид на един уеб сайт като цяло. Графичният дизайн предава

цветовите решения, шрифтове и оформление на интернет страниците на уеб сайтовете. Затова е особено важно визията на уеб сайта да е естетична и да създава впечатление за професионално комбиниране на цветове и графики. Самият уеб дизайн трябва да създава особено настроение, което следва да се асоциира с целта на уеб сайта. С една добра изработка на сайт би трябвало всеки уеб сайт да има своя уникална уеб визия и излъчване, характерно за самия уеб сайт и да е неповторимо с други уеб сайтове. Така както всеки човек си има индивидуални черти и излъчване, така и правилно изработения уеб сайт трябва да има уникална визия като графично оформление.

- Чистота на уеб дизайн: Пестеливото използване на ефекти и графични оформления създава един чист уеб дизайн, в който с лекота ще се впишат вашите материали за сайт /текстове и снимки/. С чистият уеб дизайн един уеб сайт съсредоточава вниманието на посетителя върху своята информация /услуги, стоки, продукти/. Посетителите на сайта от своя страна остават информирани и ненатоварени от излишни графики и грубо съчетаване на цветове или шрифтове. Така се увеличава процента да превърнете своите потребители в свои потенциални клиенти.

- Функционалност на уеб дизайн: Всяка една част от уеб дизайна трябва да бъде добре обмислена не само със своята визия, а и със своята функция. Така например средствата за навигация трябва да помагат на посетителя да навигира във вътрешните уеб страници и да не допускат той да се “изгуби” на сайта. Тук се говори за удобство на уеб дизайна. Добрият уеб дизайн трябва да помага на посетителите - да намерят най-бързо информацията от която се нуждаят. Само така един посетител бива задържан в уеб сайта. Графичния интерфейс на сайта трябва да спомага за подчертаване на важните рубрики в сайта, за отделните категории или под категории. Нужно е посетителя с един поглед да се ориентира кое къде и за какво служи в сайта.

Уеб дизайнерите използват различни програми, от типа на Adobe Photoshop, за да създадат оформлението и визуалните елементи на уеб сайта.

### **1.3.2. Уеб програмиране (уеб разработване)**

Уеб програмистите, наричани още уеб разработчици, вземат създадения дизайн и изработват функциониращ уеб сайт. Програмистите разбиват дизайна на неговите компоненти. След това, с помощта на различни програмни езици, те създават отделните уеб страници. Доста програмисти използват система за управление на съдържанието, с помощта на която клиентите поддържат и актуализират съдържанието на уеб сайта си.[6]

Основните технологии, които се прилагат най-често в процесът по разработка на уебсайт са:

- Писане изцяло само на програмен език (като например на HTML в Notepad)
- Използване на WYSIWYG системи, при които съдържанието по време на редактиране изглежда като крайния резултат (такива например са Adobe Dreamweaver и Microsoft FrontPage)
- Използване на програми за създаване на програмен код (като например Microsoft Visual Studio)
- Използване на системи за управление на съдържанието (като Joomla и WordPress)

Част от изброените технологии ще бъдат засегнати по-подробно в следващия раздел на настоящата глава.

## **1.4. Техники за създаване на уеб сайта**

Техниките за създаване на уеб сайт са две: създаване на отделните уеб страници с помощта на собственооръчно написан код с някой от множеството езици за програмиране или с използването на система за управление на съдържанието.

### **1.4.1. Езици за програмиране**

Същността на програмирането е да се управлява работата на компютъра на различните му нива. Управлението става с помощта на заповеди (команди) от програмиста към компютъра. Командите се издават в писмен вид



и биват безпрекословно изпълнявани от компютъра. Те са много на брой и за издаването им се използват различни видове програмни езици. Тук ще бъде представено кратко описание на възможностите на най-популярните езици:

- HTML
- DHTML
- XML
- Visual Basic
- Visual Basic Script
- JavaScript
- Java
- PHP
- C
- C++
- C#
- Python
- Ruby
- Objective-C
- ActionScript
- Perl
- CSS
- ASP
- Ajax

#### **1.4.1.1. HTML**

В първите си версии езика за форматиране на хипертекст, проектиран за простота и имащ гъвкава структура, дава възможност за визуализиране на текст и графика в Web браузър или друго приложение с подходящи възможности. Определени части от документа могат да станат хипервръзки, които при избиране предизвикват визуализирането на друга страница или секция. HTML 2 добави възможности за включване на формуляри в Web

документите, а HTML 3.2 добави тагове за таблици и тагове, които поддържат използването на JavaScript и Java. HTML 4.0 добави поддръжка на международен текст, по-голяма достъпност, по-гъвкави таблици, възможности за отпечатване и по-сложни набори от стилове, механизми за скриптове и фреймове, вложени обекти, и подобрения във формите, предлагащи допълнителни възможности за хора с недъзи. Последната известна версия на HTML, оповестена от компанията World Wide Web Consortium, е 4.01. Тя не съдържа генерални промени спрямо предната преработка, а изглажда някои от грешките, направени в нея. Поправени са и някои скриптове, за да се премахнат проблемите, които биха се появили при употребата на някои браузери. [23]

#### **1.4.1.2. DHTML**

Трите компонента на Dynamic HTML са каскадните набори от стилове (Cascading Style Sheets, или CSS), позициониране на съдържанието (Content Positioning) и downloadable fonts. Използвани заедно тези три компонента дават голяма възможност за контрол на изгледа, подреждането и определянето на поведението на web страници. [23]

##### ***Каскадни набори от стилове***

Наборите от стилове – CSS - предлагат възможност за контролиране на начина, по който HTML елементите се разполагат и визуализират. Поддръжката на CSS е разработена от World Wide Web консорциума и е въведена в HTML 3.2; допълнителна CSS поддръжка е добавена след това в HTML 4. Определението каскадни се отнася до възможността да се използват множество нива на стилови набори в документа, като едно ниво на стил може да бъде използвано за дефиниране на друго.

Дефинирани са две нива на каскадни набори от стилове. CSS1 е прост механизъм за набори от стилове, който дава възможност за асоцииране на основни стилове (например шрифтове, цветове и отстояния) към HTML елементите. CSS2 е построен върху CSS1 и добавя поддръжка на набори от стилове, специфични за различните медийни обекти; позициониране на съдържанието; шрифтове с възможност за сваляне; подредба на таблиците;

интернационализация; автоматични броячи и номериране; и други възможности.

### ***Позициониране на съдържанието***

Използвайки JavaScript, можете да промените подредбата на една Web страница динамично, и можете да модифицирате страницата по много различни начини, след като един потребител се е обърнал към нея. Съдържанието може да изчезне или да се появи, цветовете на отделните части на страницата могат да се променят. Може да се включи анимацията в web страниците чрез движение или модификация на отделните части на една HTML страница. Ползвани заедно, позиционирането на съдържанието и каскадните набори от стилове, позволяват създаването на Web страници, които използват различни стилове в различни части на страницата.

### ***Downloadable fonts***

Downloadable font-овете се използват за прикрепяне на специфични фонтове към една Web страница. Като резултат, тази страница винаги ще бъде показвана с фонтовете, които сте избрали за нея.

### **1.4.1.3. XML**

Разширяемият език за форматиране, XML - eXtensible Markup Language, все по-често присъства в живота на програмистите. XML не е език в смисъла, в който са езици Visual Basic, или C++. По-скоро той представлява набор от правила, дефиниращи как да форматираме данни или документи. Въпреки че и XML и HTML използват тагове, те се различават помежду си. Основната разлика е в това, че XML описва структурата на текста, а не външния му вид [23]. XML няма определени тагове - вместо това потребителя сам ги дефинира. Програмата е отговорна за интерпретацията на тези тагове. Именно тук идва и силата на XML, както и неговия основен недостатък. [5]

### **1.4.1.4. Visual Basic**

Това е език за разработка на приложения. Наследник е на оригиналния Basic (Beginner's All-Purpose Symbolic Instruction Code – символен код от инструкции с общо предназначение за начинаещи).

Същност на езика: интегрирана система за проектиране (създаване) на програми, управлявани от събития, включваща: текстов редактор, транслятор от смесен или компилативен тип, програма за проверяване и поправяне (дебъгер), рисуване на елементите на ГПИ, вместо тяхното текстово записване; език за програмиране, чрез който се описва реакцията на възникващите събития. [10]

#### **1.4.1.5. Visual Basic Script**

Visual Basic Script, е базиран на Visual Basic - силно ориентирана към Windows версия на Microsoft на езика за програмиране BASIC, но се използва по аналогичен начин на JavaScript. VBScript се използва за приложенията, разработени от Microsoft, ASP – Active Server Pages.

Предназначението на Visual Basic е съществено да се разшири HTML за създаване на по-динамични Web страници. Като се възползват от предимствата на конкретната платформа, тези инструменти могат да се изпълняват по-бързо и да правят повече неща от един общ пакет, предназначен за всички платформи. Но ако ги използвате, вашият сайт няма да бъде достъпен за всички посетители, тъй като не всеки притежава хардуера или софтуера, за който те са предназначени. Някои посетители няма дори да са в състояние да визуализират графика. Visual Basic Script-а се използва и от сървърната и от брауърната страна. [25]

#### **1.4.1.6. JavaScript**

Един от най-често използваните скриптов езици е JavaScript на Netscape. JavaScript е мощен език за програмиране за World Wide Web. Той не само осигурява възможност за разработване на истински интерактивни Web страници, но представлява и основното средство за интегриране на Java аплети, ActiveX контроли, plug-in модули за брауъри, сървърни скриптове и други Web обекти, позволявайки на разработчиците да създават разпределени приложения за работа в Интернет или в корпоративните интранет мрежи. Може да се използва в множество брауъри и се счита за основен в разработката на

интерактивни или анимирани уеб функции. Освен това, той се използва и в разработката на игри и писането на приложения за персонални компютри. [25]

JavaScript поддържа изпълнение на скриптове както в Web браузъри, така и в Web сървъри. Браузърните скриптове се използват за създаване на динамични Web страници, които са по-интерактивни и по-силно интегрирани с plug-in модулите, ActiveX компонентите и Java аpletите. Скриптовете могат да бъдат използвани за създаване на HTML елементи, които се добавят към визуализирания документ, или за дефиниране на функции (наречени функции за обработка на събития), които реагират на действия от страна на потребителя, като щракване с мишката или въвеждане от клавиатурата. Скриптовете могат да се използват също и за контролиране на plug-in модулите, ActiveX компонентите и Java аpletите. От страната на сървъра, JavaScript се използва за по-лесно разработване на скриптове, които обработват данни от формуляри, изпълняват търсене в бази данни и имплементират специфични Web приложения. [23]

#### **1.4.1.7. Java**

Java е съвременен обектно-ориентиран език за програмиране с общо предназначение. На Java се разработва изключително разнообразен софтуер: офис приложения, уеб приложения, настолни приложения, приложения за мобилни телефони, игри и много други. Java е език от високо ниво, който прилича на C# и C++.

Java програмите са обектно-ориентирани. Те представляват съвкупност от дефиниции на класове, които съдържат в себе си методи, а в методите е разположена програмната логика. Езикът Java е първоначално разработен и поддържан от Sun Microsystems, но през 2006 г. Java платформата става проект с отворен код. По-късно, през 2010 г. Oracle купуват Sun и в момента Java платформата се поддържа и развива от световната Java общност начело с Oracle. Езикът Java се разпространява заедно с платформата Java – специална среда, върху която се изпълнява Java кодът, наречена Java Runtime Environment (JRE). Тази среда включва т. нар. Java виртуална машина (JVM) и

пакет стандартни библиотеки, предоставящи базова функционалност за Java разработчиците. Благодарение на нея Java програмите са преносими и след като веднъж бъдат написани, след това почти без промени могат да работят на стотици хардуерни платформи и операционни системи.[3]

Минуси на езика Java:[16]

- Необходима е инсталацията на Java Virtual Machine за да може вашите приложения да вървят
- Виртуалната машина отнема допълнително процесорно време (незабележимо малко в настоящето)

Плюсове на езика Java:[16]

- веднъж компилиран код може да тръгне на всяка машина, на която има инсталирана JVM
- Възможно е използването на компилатор до машинен код, като така виртуалната машина вече не е необходима за стартиране на приложението. Така обаче се губи мултиплатформеността
- Максималната памет, която може да използва вашето приложение се заделя при стартирането му. Ако този лимит бъде достигнат не се наблюдават непредвидими странични ефекти (например в C++) - просто приложението се терминира (хвърля OutOfMemoryException)
- Няма нужда сами да се грижите за освобождаването на ресурси, които вече не използвате, Garbage Collector се грижи за това вместо вас (може да бъде извикан ръчно, въпреки че периодично се стартира сам)

#### **1.4.1.8. PHP**

PHP е най-популярния скриптов език за създаване на динамични уеб сайтове. За да може да се използва PHP е необходим уеб сървър и сървър за база данни. PHP е гъвкав, динамичен език, който поддържа множество техники за програмиране. Той има много пълен набор от възможности за обектно-ориентирано програмиране, включващо поддръжка на класове, абстрактни класове, интерфейси, наследяване, конструктори, клониране, изключения и други. [20]

PHP може да се използва на всички по-известни операционни системи, в това число Линукс, много Юникс варианти (включително HP-UX, Solaris и OpenBSD), Microsoft Windows, Mac OS X, RISC OS, а вероятно и други. Също така PHP поддържа по-голяма част от съвременните уеб сървъри. Това включва Apache, Microsoft Internet Information Server, Personal Web Server, сървърите Netscape и iPlanet, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, и много други. За по-голяма част от сървърите PHP разполага с модули, а за другите поддръжайки CGI стандарта, PHP може да работи като CGI процесор.[9]

#### **1.4.1.9. C**

Езикът C е програмен език, с който могат да се изпълняват различни задания. Той се асоциира тясно с операционната система UNIX, тъй като беше разработен на такава система. Както системата, така и повечето програми, които работят на нея, са написани на C. Обаче езикът не е свързан с нито една операционна система или компютър; и въпреки че се нарича "системен програмен език", тъй като е много удобен за писане на компилатори и операционни системи, той се използва еднакво добре и за създаване на големи програми на различните домейни. Той осигурява градивните елементи на други популярни езици като C#, Java, JavaScript и Python. Езикът се използва предимно в операционни системи и вградени приложения.[1]

#### **1.4.1.10. C++**

C++ е език за обектно-ориентирано програмиране. Създаден е от Бърн Страуструп от AT&T Bell Laboratories в края на 1985 година. C++ е разработен, като разширение на езика C, но в него са добавени класове, множествено наследяване, виртуални функции, overloading, шаблони (templates), обработка на изключения (exceptions) и вградени оператори за работа с динамична памет. Една голяма част от приложните програми на много операционни системи, както и някои от самите операционни системи, са написани на този език. Базиран на C++ са водещи програми като Firefox, Winamp и Adobe. Той се

използва за разработване на системен софтуер, приложения, сървъри с висока производителност, клиентски приложения и видеоигри.[18]

#### **1.4.1.11. C#**

Езикът е разработен от Microsoft, като част от инициативата .NET. Той комбинира принципите от C и C++ в език с общо приложение, използван за разработване на софтуер за платформи на Microsoft и Windows. C# е препоръчваният език за програмиране за .NET Framework. Архитектите на езика специално са го проектирали за .NET Framework и са го съобразили с особеностите на платформата още по време на дизайна. C# наследява простотата на Java, мощността на C++ и силните черти на Delphi. Той притежава максимално стегнат и ясен синтаксис.[4]

#### **1.4.1.12. Python**

Python е интерпретиран, обектно ориентиран език за програмиране създаден от Guido van Rossum. Той е много преносим и има налични интерпретатори за почти всички операционни системи. Python съчетава забележителна мощ с много ясен синтаксис. В него има модули, класове, разширения, типове данни от високо ниво и динамично задаване на типове. Има интерфейси за много системни функции и библиотеки, към различни графични среди (X11, Motif, Tk, Mac, MFC, wxWidgets). [21]

Python е много платформен и може да се изпълнява под Unix, Windows, OS2, Mac, Amiga и много други. Ползва се и за мобилни приложения и сайтове. Счита се за сравнително лесен език за начинаещи, защото е четим и има компактен синтаксис. Това означава, че разработчиците могат да използват по-малко редове код, за да изградят концепция, в сравнение с други езици. На базата на Python са изградени платформи като Instagram, Pinterest и Rdio. Свързаната с него програмиста платформа Django се използва от Google, Yahoo! и НАСА.[25]



### **1.4.1.13. Ruby**

Ruby е интерпретируем, интерактивен, обектно-ориентиран език за програмиране. Комбинира черти от много други езици, сред които Smalltalk, Perl, Lisp и Python. Основната му имплементация е безплатна и с отворен код. Ruby е считан за гъвкав език поради факта, че програмистите могат свободно да променят всяка една част от кода. Основни части могат да бъдат премахвани или модифицирани. Силата на езика идва от опита за премахването на ограниченията, познати в другите езици. [22]

Динамичен, обектно-ориентиран скриптов език за разработка на сайтове и мобилни приложения. Ruby е проектиран да бъде прост и лесен за писане. Той стои в основата на платформата Rails, която се ползва в Scribd, GitHub, Groupon и Shopify.[25]

### **1.4.1.14. Objective-C**

Това е обектно-ориентиран език с общо приложение. Създаден от Брад Кокс за корпорацията StepStone през 1980 година, а през 1988 е възприет като език за програмиране на NEXTSTEP, а през 1992 започва да се разпространява под GNU лиценз заедно с компилатора GCC. Той обаче се използва основно от Apple и е в основата на нейните Apple OS X и iOS, както и на APLs. Езикът може да се използва за създаването на приложения за iPhone.[12]

### **1.4.1.15. ActionScript**

ActionScript е скриптовият език на Macromedia Flash MX. Скриптовият език е начин за комуникация с програмата, при който можете да окажете на Flash какво да направи или да го попитате какво става докато филмът върви. Този двустранен начин за комуникация ни позволява да създаваме интерактивни филми. Осигурява елементи като действия (actions), оператори и обекти, които ние обединяваме в скриптове и казват на нашия филм какво да прави. ActionScript е обектно-ориентиран, интерпретируем език за програмиране. Вградените класове се наричат обекти. Movie clip са най-важните обекти във Flash филм, защото те са мини Flash филми. Потока на

изпълнение на скрипта е от първото изявление, последователно следващите до края.[15]

В ActionScript вие можете да пишете инструкции, които да заставят Flash да реагира на натиснат от потребителя клавиш или бутон на мишката по време на гледане на анимацията и да поиска определена информация от потребителя и после да отреагира на нея или можете да давате команда на Flash да намери определен кадър и да започне възпроизвеждането на филма от там. Можете да използвате ActionScript и за да анимирате обекти във вашия филм или да създадете сложни интерактивни елементи.[17]

#### **1.4.1.16. Perl**

Perl е универсален език за програмиране, създаден от Лари Уол през 1987 г. Perl е интерпретаторен език. Това означава, че кода който ще напишете се изпълнява директно, без да е необходимо компилиране, както е при C например. При съвременните езици разликата между компилиран и интерпретиран код не е вече толкова голяма. Perl е модулен език. Това означава, че той може да се разширява благодарение на модули. Модула представлява код , който може да се вгражда във вашата програма, като по този начин добавя нови възможности. Най-силните му черти са регулярните изрази (англ. regular expression, често съкращавано на regex), вградените сложни структури от данни (обикновени и асоциативни масиви) и един от най-големите в света набори от свободно достъпни модули CPAN.[11]

На Perl може да се пише процедурно, обектно-ориентирано и функционално (поддържа затваряния, познати още като обвивки, една от най-мощните абстракции в компютърната наука). Perl е слабо типизиран език. Той е интерпретиран, като програмата първо се компилира до машинно независими инструкции (байт код), които се изпълняват от интерпретатора. Интерпретаторът на Perl е написан на C и е преносим на огромен брой платформи и операционни системи.[19]

#### **1.4.1.17. CSS**

CSS (Cascading Style Sheets) е най-често използвания език за управление изгледа на интернет страници. По принцип CSS може да се използва най-общо за XML документи, но най-честата употреба е съвместно с HTML за задаване на стилове в интернет страниците. Стилите, указани чрез CSS, определят как ще изглеждат на екрана елементите на един HTML документ (интернет страница).

CSS се наричат каскадни, понеже стилове могат да указват браузърите, потребителите и авторите на интернет страниците. Тези стилове се наслаждат, в известен смисъл преливат един в друг, откъдето и термина "каскаден". CSS спестяват много работа: дизайнът на всички интернет страници на един сайт може да се зададе само с един CSS файл. CSS дават голяма гъвкавост при дизайна на интернет страниците: при нужда от смяна на дизайна е достатъчно да се промени само този единствен CSS файл. Като цяло CSS е много добре поддържан от всички широко употребявани браузъри.[8]

#### **1.4.1.18. ASP**

ASP (Active Server Pages) Технология за създаване на Интернет приложения, позволяваща да се отделят задачите на Web дизайна от задачите, свързани с реализацията на функционалността на приложението. Основна идея на ASP е създаване на Web страница с внедрени в нея фрагменти от код на скриптов езици, изпълняван от Web сървър. Технологията е разработена от Microsoft и позволява лесно да се разработват приложения за Web. ASP не се явява език за програмиране. Това е технология за предварителна обработка, позволяваща да се включват програмни модули при създаване на Web-страници. Популярността на ASP е основана на простотата при работа с езиците за сценарии (VBScript или JScript) и възможността за използване на външни COM-компоненти. ASP работи на платформа на операционните системи от семейството Windows NT и на Web сървър IIS (Internet Information Server). За разлика от скриптовите езици използвани за разширяване функционалността на брауъра, указаните фрагменти от код се интерпретират

не от брауъра, а от предназначена за целта ISAPI-библиотека, влизаща в състава на IIS. Външния фрагмент от код се замества с резултата от неговото изпълнение, като получената динамична страница се предава в брауъра. [7]

Технологията ASP получава своето развитие във вида ASP.NET — нова технология за създаване на Web приложения, базирана на платформата Microsoft .NET. Явява се ключова технология в архитектурата Microsoft .NET Framework. Основно отличие на тази технология от ASP се заключава в това че фрагментите от код в Web-страниците не се интерпретират, а се компилират и кешират, което значително ускорява работата на приложението.[7]

#### **1.4.1.19. Ajax**

Ajax (съкращение на Asynchronous JavaScript and XML) е похват в уеб разработките за създаване на интерактивни уеб приложения. Предимството на Ajax е, че посредством използването му уеб страниците се зареждат по-бързо. Посредством асинхронен обмен на малки порции данни „зад кадър“ могат да се променят само частично информации на уеб страницата. По този начин се намалява количеството информация, която се трансферира между сървъра и клиента. Асинхронността позволява да не бъде необходимо да се презарежда цялата страница отново. По този начин се повишава интерактивността, скоростта и функционалността на страниците.

Сам по себе си Ajax не е технология, а термин за употребата на група технологии, за създаване на по-добри, по-бързи и по-динамични уеб страници. Ajax-техниката се използва като комбинация от:

- XHTML (или HTML) и CSS за форматиране на текста.
- DOM (Document Object Model) реализиран чрез програмиране от страна на клиента с помощта на езици като JavaScript и JScript за динамична визуализация и интерактивност на предоставената информация.
- XMLHttpRequest (XHR) обекта за обмяна на данни асинхронно с уеб сървър. В някои случаи се използва IFrame обекта вместо XMLHttpRequest.

•XML в някои случаи е използвания формат за трансфер на данни между сървъри клиент. Все пак всеки формат е удачен, включително преформатиран HTML, чист текст, JavaScript Object Notation (JSON) и др.

#### **1.4.2. Системи за управление на съдържанието**

Системата за управление на съдържанието или Content Management System (CMS) 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. Системата за управление на съдържанието улеснява съвместното създаване на документи от различни потребители.

CMS е компютърно приложение, чрез което потребителите извършват създаване, редактиране, управление, търсене, и публикуване на различни видове съдържание. Това приложение често се използва за публикуване, съхранение, контролиране и обновяване на съдържанието във вид на новини, статии, икономически субекти, наръчници, технически ръководства, маркетингови брошури и др. Управляваното съдържание може да включва компютърни файлове, изображения, аудио файлове, видео файлове, електронни документи и друг тип веб съдържание. Най-често тези системи дават възможност на много редактори да работят по различни части на съдържанието, по едно и също време или един след друг.[24]

Веб базираната система за управление на съдържанието обхваща цялостния жизнен цикъл на веб страниците на сайта – от осигуряването на прости инструменти за създаване на съдържанието, до публикуването му и архивирането му. Тя също осигурява възможността за управление на структурата на сайта, външния вид на публикуваните страници и навигацията, осигурена за потребителите.[24]

Целта на такива системи е да улеснят изграждането на динамичен веб сайт с възможности за лесна промяна в съдържанието му по всяко време, не

само от програмисти - уеб дизайнери, а и от хора без специализирани технически знания.

На практика КМС се използват предимно за съхраняване, контролиране и публикуване на специфични документи, като новини и съобщения, упътвания, рекламни материали и др. Посредством употребата на системи за управление на съдържанието е възможно:[2]

- Лесно да бъде изработен уеб сайт, който на практика е технически издържан.
- Управление на публикуваната в сайта информация.
- Лесно и бързо публикуване на документи и материали.
- Идентификация на потребителите и разпределянето им в групи с различни права за достъп.
- Възможност да се дава определено ниво на достъп до категории от съдържанието или до отделни документи в него.
- Онлайн съхранение на всякакви видове материали (снимки, видео, документи, програми и др.).

Всяка КМС е снабдена с:

- Автоматизирани шаблони за създаване на стандартния изход (което дава вид, че цялото съдържание се съдържа на едно място);
- Инструменти за текстообработка;
- Самостоятелни модули, разширяващи функционалността на системата;
- Инструменти за управление на работния процес.

КМС предлагат широк спектър от възможности, които често ги правят предпочитано средство за реализация на уеб сайт както от клиентите, така и от разработчиците. В следващата таблична форма са представени основните предимства, произхождащи от използването на КМС.[2]

Таблица 1.1. Предимства при използване на КМС

<b>Предимства за клиента</b>	<b>Предимства за разработчика</b>
Системите за управление на съдържанието почти винаги са написани добре и са широко използвани от потребителите.	Грешки и кодиране на бъгове рядко са сериозен проблем при избор на надежден и способен доставчик.
Лесна инсталация на системата.	Улеснена администрация.
Възможност за добавяне или редактиране на страници на собствения уеб сайт.	Налице е активна общност от разработчици, чрез която повечето проблеми могат лесно да бъдат разрешени.
Спестени разходи по разработка, поддръжка и актуализация на системата.	Възможности за развитие и разрастване на функционалностите чрез използване на готови модули.
Позволява въвеждане на множество потребителски акаунти, имащи различни привилегии за достъп.	Възможност да бъдат осигурявани разработени от трети страни модули и шаблони.
Организира работния процес.	Добра SEO оптимизация.

В следващата таблична форма са представени онези положителни и отрицателни страни на системите за управление на съдържание, които могат да рефлектират върху избора на софтуерно решение за реализация на проекта по създаване на уеб сайт.[2]

Таблица 1.2. Основни предимства и недостатъци на КМС

<b>Предимства</b>	<b>Недостатъци</b>
Намаляване на времето и разходите за разработка, поддръжка и актуализация на уеб сайт.	Модификацията на системата и графичното ѝ оформяне изискват познаване на програмния (ите) език (ци), с които тя борави.

Ефективно управление на информацията, не изискващо специални технически умения.	Разширяването на функционалностите често рефлектира върху ефективността на системата като цяло.
Възможност за персонализиране и поддръжка на голям брой функции.	Ниски нива на сигурност (особено при безплатните продукти).
Наличие на вградени механизми за оптимизация на търсещи машини (SEO).	
Възможности за управление на работния процес и поддръжка на множество потребители с различни привилегии.	

Въз основа на направения обзор могат да бъдат представени следните обобщения:

- КМС позволява на потребителя да създаде или измени уеб страница, без необходимостта от необходимите технически умения.
- КМС се основава на серия от шаблони, които образуват структурата на един сайт и се използват за създаване на съдържание.
- Контент мениджъра вмъква съдържание в използваните от КМС шаблони, които след това се публикуват на сайта.
- КМС поддържат база данни за съхраняване на съдържание и метаданни, съдържащи специална информация, необходима на системата при осъществяване на индексирание.
- Съдържанието се съхранява в XML формат, което позволява гъвкавото му повторно използване.



## РЕЗУЛТАТИ И ИЗВОДИ ОТ ПЪРВА ГЛАВА

Изработката на един добър уеб сайт е процес на създаване на уникална интернет визия, изграждане на уеб страници съчетани с добре подбрана информация, функционалност и атрактивност.

В първа глава бяха разгледани видовете уеб сайтове от гледна точка на тяхното предназначение и приложена технология, бяха разгледани под процесите уеб дизайн и уеб програмиране на процеса за създаване на уеб сайт. Също така бяха разгледани техниките за създаване на уеб сайт: чрез писане на собствен код с помощта на един или няколко езика за програмиране, като беше дадена обобщена информация за по често използваните от тях или чрез използване на системи за управление на съдържанието, като при тях бяха разгледани положителните и отрицателните им страни.

След направения анализ и въз основа на дадената информация като заключение може да бъде казано, че системите за управление на съдържанието (КМС) са най-подходящият избор за направата на уеб сайт за детска градина. Основната причина довела до този извод е, че КМС дава възможности за лесна промяна в съдържанието на сайта по всяко време, не само от програмисти - уеб дизайнери, а и от хора без специализирани технически знания. Нуждата от тази възможност идва от едно от изискванията поставени за направата на конкретния сайт за детска градина, а то е персоналът на детската градина сам да може да добавя и променя информация в него.

Предимствата на системите за управление на съдържанието, както и възможността те да бъдат осигурени безплатно и с отворен код дават основание да се твърди че именно те са технологията, която може да бъде разглеждана като предпочитана при изграждане на уеб сайт, който ще бъде управляван от неспециалисти.

## ГЛАВА ВТОРА: СПЕЦИФИКИ НА КМС ОТ ГЛЕДНА ТОЧКА НА ВЪЗМОЖНОСТТА ЗА ИЗГРАЖДАНЕ НА САЙТ ЗА ДЕТСКА ГРАДИНА

В днешно време съществуват стотици КМС и за направата на правилния избор коя то тях е най-подходяща за създаване на сайт за детска градина трябва първо да се запознаем с тях. В тази глава ще бъдат разгледани накратко структурата и видовете КМС.

### 2.1. Структура на КМС

Елементите, съставляващи всяка система за управление на съдържанието могат да бъдат сведени до три основни: [2]

- Система за събиране;
- Система за управление;
- Система за публикуване.

Най-общо казано, системата за управление на съдържание служи за събиране, управление и публикуване на парчета от информация. На входа на системата постъпва сурова информация, която попада директно в *системата за събиране*. В процес на обработка, тази сурова информация постъпва в определена категория на съдържанието и бива манипулирана от системата за управление. *Системата за управление* може да бъде представена като база данни, съхраняваща тези парчета информация. *Системата за публикуване* има за задача да извика определен елемент от категорията, намиращ се в системата за управление и да го превърне в публикация. Въпреки, че логически са отделени, трите елементи на системата често се припокриват. Така например:[2]

- *Системата за управление може да служи като част от системата за събиране* – подобна ситуация е налична когато даден контент попада в хранилището (и съответно в системата за управление) преди да е напълно завършен.

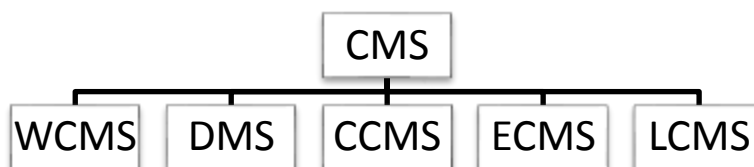
- *Системата за управление може да служи като част от системата за публикуване* – хранилището съхранява съдържание, което се

извиква при заявка от потребителя. Така управлявания контент е част едновременно и на двете системи.

- *Системата за публикуване може да служи като част от системата за събиране* - След като бъде създадено съдържанието то попада в хранилището. За да попадне там, обаче, е необходимо автора му да въведе информацията в предложена от системата форма (шаблон). При заявка на конкретния контент от потребителя на уеб сайта, системата за публикуване създава уеб формата, която системата за събиране използва.

## 2.2. Видове системи за управление на съдържанието

Видовете контент мениджмънт системи, получени като резултат от деление според типа на управлявания контент са представени във Фигура 2.1., а в Таблица 2.1. са представени кратки описания за всеки един от видовете.[2]



Фигура 2.1. Видове КМС

Таблица 2.13. Видове контент мениджмънт системи

Вид	Пълно наименование	Описание	Примери
<b>CMS</b>	<i>Система за управление на съдържание (Content Management System)</i>	Термин, който се използва за да обобщи всеки от посочените в следващите редове видове системи.	
<b>WCMS</b>	<i>Система за управление на уеб съдържание (Web Content Management System)</i>	Програма, която позволява да бъде създавано, редактирано и публикувано съдържание в уеб сайт.	WordPress, Joomla, Drupal, Plone и мн. др.
<b>CCMS</b>	<i>Компонентна система за управление на съдържанието</i>	Използва се за управление на съдържанието на ниво на компонент, а не на ниво документ. Всеки	Vasont, XDocs, easyDITA, DITAToo,

	<i>(Component Content Management System)</i>	компонент представлява една единствена тема, концепция или активи (например изображение, таблица, описание на продукта, процедура).	SDL LiveContent и др.
<b>ECMS</b>	<i>Система за управление на корпоративно съдържание (Enterprise Content Management System)</i>	Интранет инструмент, който позволява на всеки в компанията, прилагаща системата да осъществява достъп, управление и преглед на документи, шаблони, различни медии и други информационни активи.	EMC Documentum, Atlassian Confluence, Oracle Web Center и др.
<b>DMS или EDM</b>	<i>Система за управление на документи (Electronic Document Management)</i>	Този тип системи служат за съхранение, достъп и управление на текстови файлове с различен формат. Понякога се разглежда като компонент на ECMS.	DocPath, Document Locator, SharePoint, Livelink
<b>LCMS или LMS</b>	<i>Система за управление на образователен контент (Learning Content Management System)</i>	Система, която позволява да бъде лесно управлявано, публикувано и получавано от потребителите учебно съдържание.	Joomla LMS, Moodle, Ilias и др.

Всяка от представените по-горе видове системи борави с различен по вид контент, спрямо другите системи, но имат една и съща цел: да осигуряват ефективно управление на информационните ресурси. При това всяка от системите включва еднотипен с другите системи набор от функции, сред които:

- Съхраняване на съдържанието;
- Контрол на достъпа до съдържанието;
- Проверка на съдържанието;
- Управление на жизнения цикъл на съдържанието;

- Търсене из съдържанието;
- Публикуване на съдържанието;
- Предоставяне на анализи и/или доклади.

Въпреки приликите по отношение изпълняваните функции, основни характеристики и фундаментална цел, обаче са видими и значими разлики между отделните видове системи. Всеки отделен вид КМС се фокусира върху различен набор от цели и задачи. Освен това съществуват разлики по отношение на възможностите, предлагани от различните КМС. Тази причина налага предварителна дефиниция на вида на съдържанието, както и начина, по който следва то да бъде управлявано преди да бъде осъществен избор на КМС. Поради тази причина в следващите редове ще бъдат поотделно разгледани спецификите на представените видове контент мениджмънт системи.

### **2.2.1. Компонент контент мениджмънт системи**

Традиционните системи за управление на съдържанието се съсредоточават главно върху управлението на метаданни и извличането на цели документи. В случаите, в които е необходим „по-гранулиран” достъп в документа се прилагат т.нар. *компонент контент мениджмънт системи* (CCMS). [2]

Гранулираният достъп е полезен в следните два случая:

- при осъществяване на връзка с определен фрагмент, намиращ се в рамките на един документ;
- при нужда от повторна употреба на части от документи в различни други издания.

С други думи CCMS има общи черти с традиционните системи, управляващи контент, но основната разлика е тази, че те не работят с цели документи, а само с части от тях. Различават се също така и по адресирането и начина на свързване на съдържанието за всяко ниво. Те управляват съдържанието на по-дълбоко ниво, като боравят с парчета от даден документ (параграф от текст, единични теми, понятия, или информационни ресурси като изображения, отделни таблици или описания на продукти), а не с всички

налични контенти. Въз основа на това може да се каже, че CCMS е специален вид КМС, която поддържа възможност за изграждане на нови публикации от повторно използвани фрагменти (части) от друга или други публикации, без да се отчита тяхното физическо ниво.

CCMS предлага ясни предимства пред традиционните КМС:[2]

- Оптимизирано повторно използване на съдържанието.
- Икономия на време и ресурси при управление на съдържанието.
- Икономия на време и ресурси, необходими за създаване на публикации от повторното използване на фрагменти от съдържанието.
- Високо качество на публикациите.
- Високо ниво на достъпност на съдържанието (достъп във всяко гранулирано ниво).

Всеки компонент поддържа данни за жизнения му цикъл, версията и нивото на одобрение. С други думи, всяка отделна част от съдържанието поддържа свой работен процес, като неговия статус (състояние) може да бъде проследено самостоятелно или като част от по-голяма система. Тези данни разрешават проследяването на отделните компоненти заедно или в група по даден признак. Характерна черта на този вид КМС е това, че тя може да бъде съставна част на по-голяма система за управление на съдържанието.

### **2.2.2. Система за управление на образователен контент**

Системите за управление на образователен контент (LCMS) са уеб базирано софтуерно приложение, подпомагащо процесите, свързани с нормалното протичане на електронната форма на обучение, доставяйки достъп до образователни курсове или програми за обучение. Освен за да доставят достъп до онлайн съдържание, този тип системи служат и за:[2]

- Автоматизиране на процесите, свързани с поддръжката на документация;
- Регистрация на потребителите;
- Администрация;

- Осъществяване на анализ на пропуските в уменията, проследяване на движения и активности, както и извършване на отчетна дейност.

Една LCMS трябва да бъде в състояние да осигури:[2]

- Централизирана и автоматизирана администрация;
- Самостоятелно ръководство на услуги;
- Доставка на учебно съдържание;
- Осигуряване на единни стандарти;
- Персонализация на съдържанието, даваща възможност за повторно използване на знания.

LCMS е софтуер, който осигурява среда за съвместна работа на много потребители, всеки от които има различни привилегии за достъп и работа с постъпилото в системата съдържание. Основните предимства на подобен тип система, произхождат от предимствата на традиционните КМС и са свързани с това, че потребителите могат да създават и повторно да използват съдържание за електронно обучение при същевременно намаляване на дублирането на създаваните контентни. При това, съхраняваните обекти могат да бъдат направени достъпни, както за групата, изучаваща съответната дисциплина, така и за експертите, отговорни за организирането и управлението на представяното съдържание. [2]

Няколко са причините, които могат да окажат допълнително въздействие за включването на LCMS в дейността на работещите посредством традиционните методи за обучение организации:[2]

- *Минимизиране на разходите* - тъй като обучението се осъществява посредством интернет (вместо в традиционна класна стая) не се генерират разходи от наем на помещение или субсидиране на пътни разходи на преподавателите. В допълнение, цената на курсовете за електронно обучение са около 50 на сто или по-малко от цената на обучението, провеждано по традиционен начин, което допълнително облекчава разходния план.

- *По-бързо и по-ефективно предоставяне на информацията* в сравнение с традиционната форма на обучение – това предимство става по-

осезаемо при наличието на големи групи обучавани. В това отношение уеб базираната среда съдейства в две направления: от една страна съдейства за бързото доставяне на съдържание, а от друга предоставя възможност то да бъде бързо и лесно актуализирано.

- *Гъвкавост по отношение времеви интервали и място за провеждане на обучението* – наблюдава се липса на необходимост обучаваните да посещават на място обучителните сесии, да спазват твърд график или да пътуват дълги разстояния за да бъдат част от обучаваната група.

### **2.2.3. Система за управление на корпоративно съдържание**

Най-общо казано ЕСМ е бързо развиваща се област, включваща работа с приложения като електронна поща, мигновени съобщения, бизнес комуникация, записи, снимки, търговска информация, контакти, цифрово управление на активи. ECMS се явяват комплексно решение, което осигурява:[2]

- Събиране и систематизация на неструктурирана информация;
- Поддръжка на бизнес процеси;
- Възможности за анализ на информацията.

Елементите участващи в една система, управляваща корпоративна информация са:[2]

***Управление на електронни документи (Integrated Document Management, IDM):*** IDM системата по същество представлява електронна библиотека, имаща следните функции:

- Съхранение на документи и метаданни;
- Разграничаване на достъпа;
- Водене история на работа с документите.

***Управление на записите (Records Management, RM):*** RM изпълняват следните функции:

- Водене на архиви;
- Регистрация на входящи, изходящи и вътрешни документи;
- Организиране на деловодство;



- Поддръжка пълния жизнен цикъл на документа (една от най-важните функции на ECM).

**Управление бизнес-процеси (Business Process Management, BPM) и управление на работен процес (Workflow):** Ролята на BPM в концепциите за пълно ECM решение е значителна. Без интеграция на бизнес процеси, всяка система остава статична и служи само като хранилище на информация. Реална полза от внедряването на ECMS може да се получи само при интеграцията ѝ с основните бизнес системи на организацията. Ето защо може да се твърди, че основно предназначение на ECM е включването на неструктурираната информация в бизнес процесите. BPM и Workflow не могат да бъдат разглеждани като елементи на ECM. Те по-скоро представляват свързващо звено между системите ECM и системите за обработка на трансакции.

**Сътрудничество:** Средствата за организация на сътрудничество позволяват да бъде осъществявано взаимодействие в случаите, които не се поддават на строга формализация (във всяка задача, която е свързана с творческа работа или присъства процес на преговори). Тези системи притежават достатъчна функционалност по отношение управлението на документи от една страна и възможности за управление на процеси от друга.

**Управление на знания (Knowledge Management, KM):** Основни проблеми за системите, управляващи корпоративно съдържание, чието решаване изисква помощта на системи за управление на знания са свързани с:

- Големите обеми информация, изискващи мощни търсеци механизми;
- Осигуряването на прозрачен достъп към разнородни източници на информация;
- Автоматичната категоризация на информацията по съдържанието на документите;
- Удобство при визуализация на големи масиви информация.

Предложените от ECMS решения на коментираните проблемни области са посочени по-долу:

- Пълно текстово търсене – всички ЕСМ решения притежават вградени търсещи машини;
- Пълзящо търсене по разнородни източници на информация;
- Автоматична категоризация;
- Прозрачна интеграция в интегрирана комплексна система;
- Визуализация на информацията.

**Корпоративен портал:** Порталът е приложение, което предлага единен достъп както до външна за организацията информация, така и до вътрешна персонализирана информация с цел вземане на стратегически бизнес решения. Порталът формално не се явява част от ЕСМ, но без него е невъзможно да се оформи представа за системата като цяло.

Въз основа на представената до тук информация може да се твърди, че ЕСМС са резултат от сближаването на четири подхода, а именно:

- Управление на уеб съдържание (WCMS);
- Електронно управление на документи (EDMS);
- Управление на цифрови активи (DAMS);
- XML публикуване.

ЕСМС е система, използваща набор от процеси, инструменти и стратегии за обработка на съдържание, активи, документи и друга информация, която определя структурата и йерархията на организацията.

#### **2.2.4. Система за управление на документи**

Електронната система за управление на документи (EDMS) е софтуер, който управлява различните етапи от жизнения цикъл на електронните документи. Продуктите, с които работи една EDMS са електронни интерпретации на физически документи, които често включват статични изображения и графики. Иначе казано, EDMS управляват както цифрови документи, така и дигитализирани версии на книжни документи. Този тип системи са предпочитани от потребителите при процеси, свързани с поддръжката и управлението на документооборота. Причина за това е факта, че

една EDMS взема отношение по множество рутинни въпроси, част от които са посочени в Таблица .:[2]

Таблица 2.2. Въпроси, решавани от една EDMS

<i>Показател</i>	<i>Поддържани възможности</i>
<b>Местоположение</b>	<p><i>Къде и как ще бъдат съхранявани документите?</i></p> <p>EDMS предлага възможности за локализиране записите на даден файл, аналогични на тези при работата с графичен потребителски интерфейс на операционна система например. Организацията на различните документи следва йерархична структура, при която се наблюдават различни равнища (нива) в организацията на съществуващите контентни (родители и наследници).</p>
<b>Наливане</b>	<p><i>Как ще бъдат създавани документите? Какви методи ще бъдат използвани при тяхната организация и индексация за да бъде облекчено извличането им в по-късен етап?</i></p> <p>Системите за управление на документи обикновено използват база данни за съхраняване на метаданни на документите и системните файлове за съхраняване на действителните физически файлове.</p>
<b>Извличане</b>	<p><i>Как ще бъдат намирани и извличани документите?</i></p> <p>Процесът, нар. „извличане” обхваща прелистването на документите и търсенето по конкретен показател.</p>
<b>Сигурност</b>	<p><i>Как ще бъдат подсиgurени документите?</i></p> <p>Системата позволява ограничаване правата за достъп, така че не всеки потребител да има пълен достъп до съхраняваните документи.</p>
<b>Възстановяване</b>	<p><i>Как да бъдат възстановени документите в случай на унищожаване от пожари, наводнения или природни бедствия?</i></p>
<b>Период на съществуване</b>	<p><i>Колко дълго трябва да бъдат съхранявани документите?</i></p> <p>При разрастване на организациите се увеличава и броя на наредбите, като неформалните указания за поддържане на различни видове документи, дават път на по-формални практики за управление на записи.</p>
<b>Архивиране</b>	<p><i>Как могат да бъдат съхранени документи за бъдеща употреба?</i></p>
<b>Разпространение</b>	<p><i>Как може да бъде подсиgurено разпространението на</i></p>

	<i>документи до всички лица, имащи нужда от тях?</i>
<b>Работен процес</b>	<i>Какъв път трябва да премине документа в своето съществуване?</i>
<b>Създаване</b>	<i>Как се създават документи?</i> Този въпрос е особено важен в случаите, в които множество потребители трябва да си сътрудничат.
<b>Автентичност</b>	<i>Има ли начин да бъде гарантирана истинността на документите?</i>
<b>Проследяване</b>	<i>Кога, къде и от кого са създадени, модифицирани, публикувани и съхранени документите?</i>

### 2.2.5. Система за управление на уеб съдържание

Системата за управление на уеб съдържание (*Web Content Management System, WCMS*) представлява програмен комплекс, предоставящ функции по създаване, редактиране, контрол и организация на уеб сайтове. WCMS често се използват за създаване на блогове, лични страници, и интернет магазини и са насочени към потребители, слабо запознати с програмирането. Този тип системи използват трислойна архитектура клиент/сървър. Получавайки заявка сървъра за приложения я обработва, свързва се с базата данни и генерира съответен резултат към клиента във вид на HTML файл. Сървърът за приложения се явява стандартизирана платформа за динамична доставка на контент и създаване на основните приложения. Сървърите за приложения могат да бъдат няколко, като връзката с тях се реализира посредством уеб сървъра.[2]

Основните възможности на WCMS са обобщени както следва:[2]

- Използване на шаблони за изобразяване, автоматично прилагани към нов или съществуващ контент.
- Простота на редактиране на контента, посредством употребата на текстов редактор (процес, при който не се изискват специфични познания по програмни езици).
- Възможност за разширяване на функционалността на уеб сайта чрез инсталиране на допълнителни модули и плъгини;

- Визуализация на контента, позволяваща преглед и редакция на контента преди публикуването му на сайта.
- Управление на документооборота.
- Единна категоризация на всички видове съдържание.
- Управление на потребители с различни нива на достъп.

### 2.2.5.1. Видове WCMS

Съществуват множество готови системи за управление жизнения цикъл на уеб сайт. Те могат да бъдат разделени на три типа в зависимост от начина им на работа в следните три вида:[2]

- Генериране на страници по заявка;
- Генериране на страници при редактиране
- Смесен тип.

*WCMS, генериращи страници по заявка:* Системите от този тип работят на базата на верига: Модул за представяне генерира страници при заявка, като използва информация от база данни, а информацията в базата данни се променя с помощта на модул за редактиране.

*WCMS, генериращи страници при редакция:* Системите от този тип се базират на програми за редактиране на страници, които при промяна на съдържанието на сайта създават набор от статични страници. При този подход липсва обратна връзка с посетителя на сайта.

*WCMS от смесен тип:* Както се разбира от названието, този тип системи съчетават в себе си преимуществата на първите два изброени подхода. Генерирането на страници тук може да бъде реализирано чрез използване метода на кеширане. Тук генерирането на страницата се осъществява еднократно, а всяко повторно зареждане се разтоварва значително по-бързо от кеша. Кешът се обновява както автоматично след изтичане на определен срок, така и ръчно след извършена команда от администратора на системата.

## 2.3. Комерсиални, Open Source и потребителски КМС

При разработване на уеб сайт с помощта на система за управление на уеб съдържание, въпрос от съществена важност е каква система да бъде избрана. Възможностите тук са три, а именно:[2]

- *Комерсиална WCMS;*
- *WCMS с отворен код;*
- *Потребителска WCMS.*




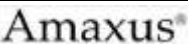

Този раздел разглежда предимствата и недостатъците на всеки подход

### 2.3.1. Комерсиални КМС

Този тип системи все още са популярен избор на КМС решение. Това се дължи основно на факта, че много потребители предпочитат да получават постоянна техническа поддръжка и обслужване, гарантирайки безопасност и сигурност при работа със системата. Налице е също така и факта, че тези системи са вече построени и отговарят на конкретни потребителски потребности. Това означава, че те могат да се приложат веднага, за разлика от системите с отворен код, които трябва да се настройват откъм функционалности например. Наред с това всеки инсталационен пакет е придружен от стабилна документация, потребителски ръководства и наръчници за обучение, всеки от които е значим както за редакторите, така и за администраторите на системата.[2]

На пазара се предлага широка гама от търговски КМС. Най-популярните от тях са представени в Таблица 2.3.:

Таблица 2.3. Примери за комерсиални КМС

<i>Наименование на Системата</i>	<i>Уеб адрес</i>	<i>Лого</i>
Sitecore	<a href="http://www.sitecore.net/#">http://www.sitecore.net/#</a>	
Episerver	<a href="http://www.episerver.com/">http://www.episerver.com/</a>	
Ektron	<a href="http://www.ektron.com/">http://www.ektron.com/</a>	
Amaxus	<a href="http://www.amaxus.com/">http://www.amaxus.com/</a>	
Sitefinity	<a href="http://www.sitefinity.com/">http://www.sitefinity.com/</a>	

### 2.3.1.1. Предимства на комерсиалните КМС

Има няколко причини комерсиалните КМС да бъдат предпочитани от потребителите: [2]

- *Стабилност* – една комерсиална система предлага по-голяма стабилност от системите с отворен код, при които се осъществява постоянно тестване и оценка.
- *Ползваемост* - комерсиалната КМС е изключително удобна за използване от потребители, които нямат технически познания поради две основни причини: от една страна системата предлага лесен и удобен интерфейс за използване, а от друга тя е придружена от потребителски наръчници и ръководства за обучение. Една комерсиална система е проектирана да се използва както от технически обучени потребители (специалисти в областта), така и от потребители нямащи специални технически познания.
- *Помощ и поддръжка* – както стана ясно, комерсиалните системи предлагат ръководства и наръчници за обучение на своите потребители. Наред с това те имат постоянен достъп до техническа поддръжка.
- *Документация* – документацията на продукта трябва да бъде добре написана и лесна за разбиране. Така документацията ще отговори едновременно както на потребителите, така и на изискванията на търсачките.

### 2.3.1.2. Недостатъци на търговските КМС

Основните недостатъци наблюдавани при търговските КМС са: [2]

- *Цена* - комерсиалните системи често се продават на висока цена.
- *Ограничения* - Поставени са ограничения върху архитектурата, интерфейс дизайна и функциите, което не позволява на потребителя да осъществи персонализация на системата. В този смисъл системата не може да се разширява. Ако даден потребител (или фирма) изисква определена функция, доставчика на системата ще направи анализ на разходите и ползите, за да провери дали производството на тази функция е осъществима. В случай, че анализа покаже, че има достатъчен интерес към исканата функция е възможно

тя да се създаде, а след това да се предлага и на други клиенти срещу заплащане.







- *Опасност от поглъщане* - в съвременните пазарни условия постоянно съществува риск друг доставчик на КМС да погълне използвания. Това означава, че доставчика на КМС, от който е закупен съответния лиценз, поддръжка и техническа помощ може да бъде закупен от друг доставчик. Това ще донесе редица неудобства за потребителите на „погълнатия” продукт, по-важните от които са свързани с липса на възможност за поддръжка, консултации и помощ.

### 2.3.2. Open Source CMS

Както може и да се предположи, най-широко разпространение имат КМС с отворен код. Системите предлагат както свободни, така и платени версии. В първия от двата варианта лицензионното споразумение позволява често безплатно ползване на софтуера както за лична употреба, така и от организации с нестопанска цел. [2]

Съществуват множество системи с отворен код. Най-популярните от тях са посочени в Таблица 2.4.2.4.

Таблица 2.4. Примери за КМС с отворен код

<b>Наименование на системата</b>	<b>Уеб адрес</b>	<b>Лого</b>
WordPress	<a href="http://bg.wordpress.org/">http://bg.wordpress.org/</a>	
Joomla	<a href="http://joomla.bg/">http://joomla.bg/</a>	
Drupal	<a href="http://drupalbg.org/">http://drupalbg.org/</a>	
Made Simple	<a href="http://www.cmsmadesimple.org/">http://www.cmsmadesimple.org/</a>	
Impress	<a href="http://www.impresscms.org/">http://www.impresscms.org/</a>	
Plone	<a href="http://plone.org/">http://plone.org/</a>	

#### 2.3.2.1. Предимства на КМС с отворен код

Няколко са основните предимства на системите с отворен код, най-съществените, от които са :[2]



- *Ниска цена* - изходният код може да бъде получен безплатно, а поддръжката, предоставена от тези системи често е с пъти по-евтина от същата при комерсиалните КМС.

- *Гъвкавост и лесна персонализация* – достъпът до кода осигурява възможности за адаптиране и персонализация на системата, така че тя да удовлетворява всички бизнес изисквания.

- *Поддръжка* - решение на всички проблеми, свързани с експлоатацията, поддръжката и персонализирането на конкретна система с отворен код може да бъде осигурено чрез единната работа на общността от разработчици. Тази общност предоставя съвети и препоръки за решаването на съществуващ проблем.

- *„Отворена” платформа* - системата с отворен код е проектирана с помощта на „отворени” езици за програмиране като Java, Python, PHP и т.н. Тези популярни софтуерни платформи предоставят достъп до постоянна поддръжка и информационни ресурси.

- *Интеграция* - способността да се разшири и персонализира системата с отворен код означава, че тя може да се интегрира с друг софтуер (например XML).

- *Незабавна резолюция за грешка* - тези системи са отворени за мненията на общността. В случай, че възникне проблем, неговото решение ще бъде скоро посочено от общността. Свободният достъп до изходния код позволява проблема да бъде решен от самия потребител.

### **2.3.2.2. Недостатъци на КМС с отворен код**

Както стана ясно по-рано, нито една система не е 100% перфектна. Някой от недостатъците на тези системи са: [2]

- *Слабост в проектирането на системата* - Много от системите с отворен код са съсредоточени върху техническите аспекти (като например структура, функции, шаблони) за сметка на използваемостта. Това е недостатък, осезаем предимно от лицата, които не притежават сериозни технически умения. Те работят основно в т.нар. „front end” панел и тяхна

основна грижа са създаването и публикуването на съдържание за минимум изразходвано време.

- *Липса на документация* – поради наличието на „подкрепа от КМС общността” много системи с отворен код предоставят малко или изобщо не предоставят документация, придружаваща системата.

- *Риск от „свърх персонализация”* - една от силните страни на системите с отворен код е способността им да бъдат адаптирани и персонализирани. Съществува опасност от наличието на постоянни промени, увеличен брой функции и прекомерно персонализиране като резултат от работата на твърде много хора, имащи достъп до изходния код. Налице е състояние, в което една система бива „осакатена” в резултат от прекомерното добавяне на нови функции, което може да предизвика поредица от проблеми в бъдеще.

### **2.3.3. Потребителски КМС**

Потребителските КМС, наричани „КМС по поръчка” и „персонализирани КМС”, се основават на рамката на системите с отворен код. Персонализирането на „нови” КМС, базирани на системи с отворен код може да съдейства за преодоляване на пропастта между Open Source и комерсиални КМС. Изграждането на потребителска КМС осигурява солидна рамка за управление на съдържанието, което дава основание за персонализация, съобразена с уникалните изисквания на конкретния клиент. Аргументите срещу поръчките КМС, подобно на аргументите срещу използването на КМС с отворен код са свързани с понятието несигурност. Персонализираните КМС предизвикват опасения, свързани с развитието на системата, което често е свързано с дефиниране на допълнителни изисквания към поддръжката на продукта. [2]

### **2.4. Избор на Open Source WCMS за нуждите на детска градина**

След направения анализ на видовете КМС и поради обстоятелството, че изграждането на уеб сайтът за детска градина на настоящата дипломна работа трябва да е на възможно най-ниска цена, трябва да бъде избрана система

за управление на съдържание, отговаряща на критериите “отворен код” и „управляваща уеб контент”. Поради тази причина в следващия раздел ще бъде направен преглед на характеристиките, функционалностите, положителните и отрицателните страни на най-популярните системи за управление на съдържание, отговарящи на тези критерии.

В настоящия раздел ще бъде направено сравнение между трите най-популярни безплатни WCMS (Wordpress, Joomla, Drupal) и една система от по-нисък клас (Plone).

### **2.4.1. CMS WordPress**

WordPress е публикувана за първи път през 2003 г. стартирайки като обикновена блог система. Подобно на всяка КМС, тя се основава на серия от уеб шаблони и служи да управлява създаването в средата ѝ съдържание.

Основното предимство, което предлага е възможността да бъде използвана от технически необучени потребители, които не притежават специфични познания по HTML или PHP например. Нещо повече – потребителите имат възможност да разширят функционалностите на създадения с помощта на системата блог, използвайки специално разработени за целта разширения. В WordPress малките програми, които добавят разширения на функционалностите се нар. плъгини.[2]

Системата WordPress може да се инсталира на всеки компютър (или интранет) и позволява на потребителя да има цялостно управление на създадения с нейна помощ блог.

Основната функция на WordPress, свързана с управление на съдържанието изисква наличие на поредица от инструменти и процеси, част от които включват: [2]

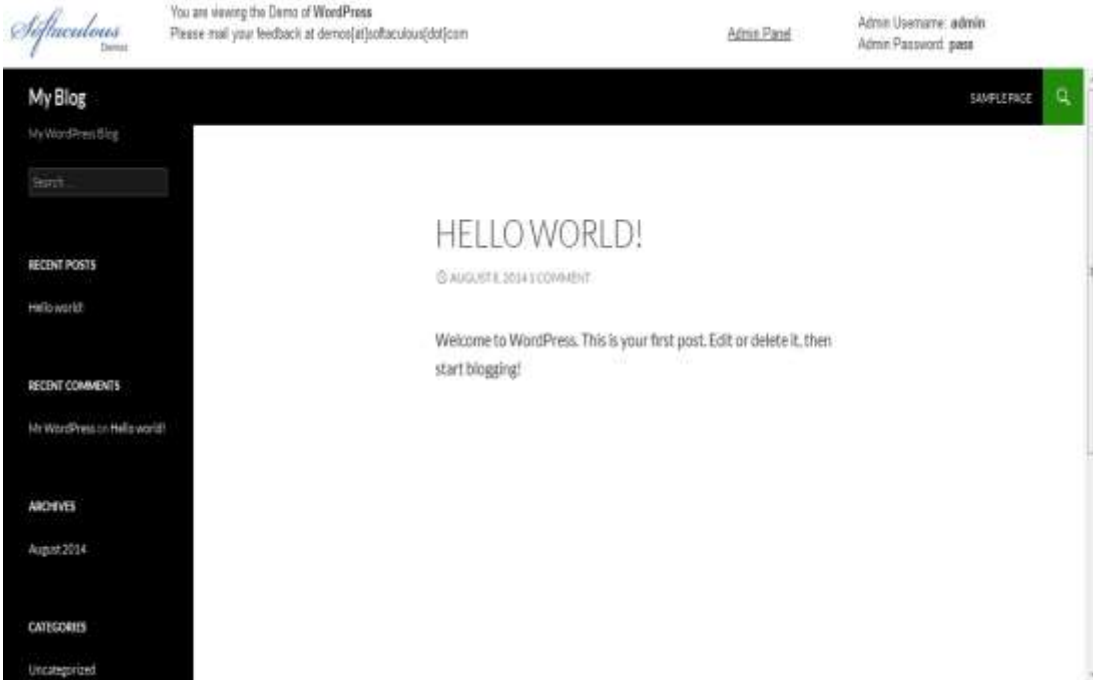
- Използване на файлов мениджър за качване на документи и изображения;
- Подсигуряване на част от съдържанието посредством изискване на парола за достъп до него;
- Организиране на съдържанието в категории и под-категории;

- Облекчено форматиране на текст посредством употребата на инструменти, предложени от специален текстов редактор;
- Поддръжка на възможности за архивиране и търсене;
- Възможност за създаване на блогрол секция;
- Възможност за поддръжка на „мултиблог“;
- Предоставяне на SEO инструменти, за постигане на максимална оптимизация.

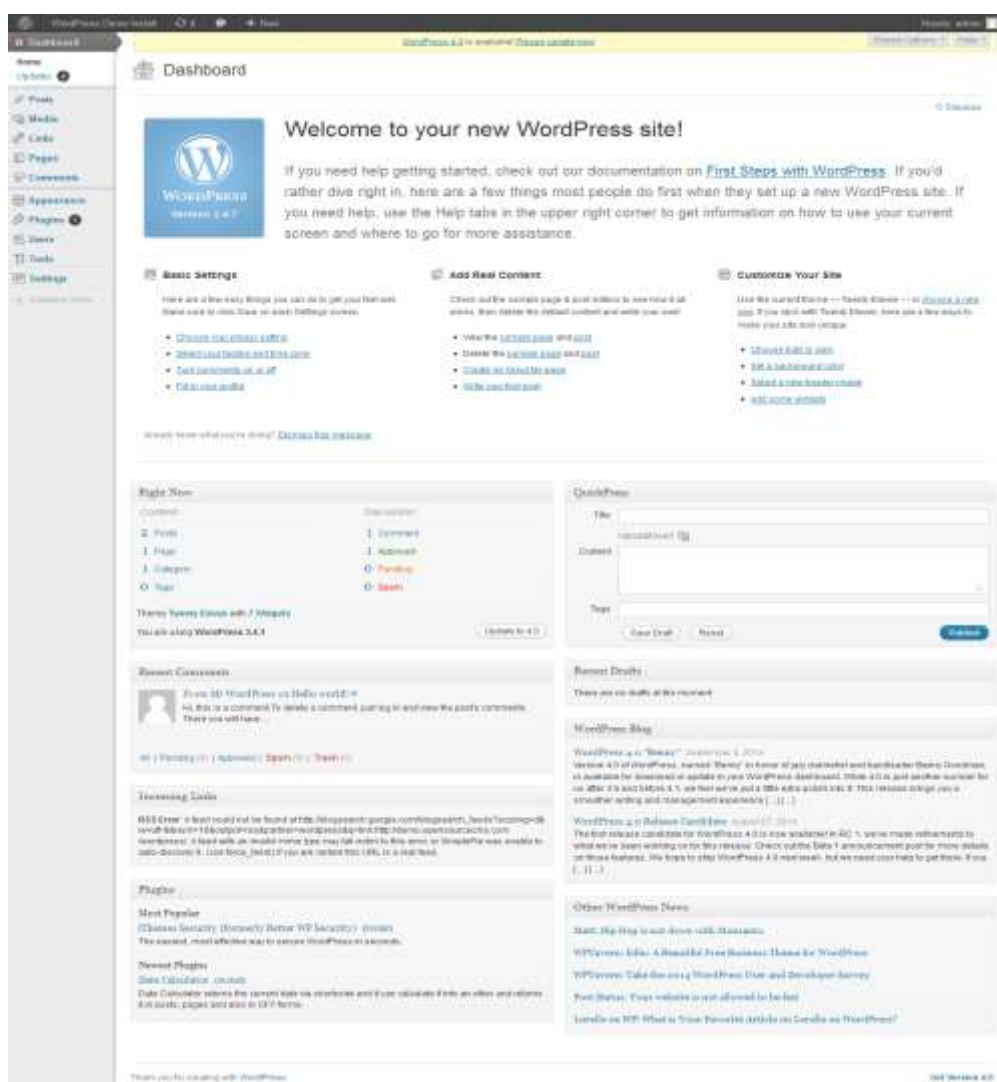
Поддръжниците на WordPress твърдят, че множеството предлагани инструменти и облекчен режим на работа са само част от положителните страни на тази система и според тях негативни коментари по неин адрес не могат да бъдат изтъкнати. Критиците на системата, обаче отразяват следните забележки: [2]

- Наличие на слабости по отношение сигурността на системата – КМС, подобно на всеки друг софтуер е податлив на атаки от хакери. Критиците считат, че този проблем може да бъде преодолян с повишаване сигурността на системата.
- Бавно зареждане на страниците – може да зависи от странични фактори, но според специалистите в сектора, кеш плъгин може да разреши този проблем.
- Начинаещите потребители срещат трудности в работата със системата – за преодоляване на този проблем, специалистите съветват следене и водене на дискусии в WordPress общността, както и използването на множеството съществуващи клипове и онлайн уроци.

На Фигура 2.2. е представен началния екран на системата WordPress в демо версията на продукта, а на Фигура 2.3. е представен достъпа до административната страница на системата при въведени параметри за вход.



Фигура 2.2. Начален екран на WordPress (Демо версия)



Фигура 2.31. Административна страница на CMS WordPress (Демо версия)

## 2.4.2. CMS Joomla

CMS Joomla е популярна система с отворен код, която е свободно разпространявана. Системата позволява лесно персонализиране и разширяване на функционалността, което според голяма част от привържениците на системата са част от нейните силни страни. CMS Joomla позволява на обикновения потребител да изгради и поддържа един или няколко уеб сайта, както за интернет, така и за интранет. Системата може да бъде използвана за разработването както на малки, така и на стабилни приложения.

Joomla е публикувана през 2005 г. и е два пъти носител на наградата за управление на съдържание Packt Open Source (през 2006 г. и 2007 г.) като част по-важните ѝ характеристики са:[2]

- Използва се за изграждането на различни по тип уеб сайтове (сайтове за електронна търговия, портали, корпоративни интранет и пр.).
- Системата е написана на PHP и позволява редакция и персонализация на кода.
- Съхранява данните в база данни MySQL и използва обектно-ориентирано програмиране при управлението на съдържанието (снимки, видео, аудио и текст).
- В Joomla малките програми, които добавят разширения на функционалностите се наричат „Extensions”, а всяко от тези разширения попада в една от групите: компоненти, модули, плъгини.
- Потребителят има възможност за избор и настройка на шаблони, всеки от които има възможност да бъде приложен както за целия сайт, така и за отделни части от него.
- Притежава няколко основни инструменти, подпомагащи управлението на различните по вид медии („Media Manager”, "Language manager”, “User manager”, “Category Manager”).
- Съдържанието се създава с помощта на WYSIWYG редактор (What You See Is What You Get), а се редактира, организира и публикува по същия начин, както при всяка друга КМС.

Привържениците на системата дефинират следните предимства:[2]

- Възможност да бъдат създавани многоезични уеб сайтове;
- Лесна за инсталация, употреба и поддръжка;
- Висока степен на гъвкавост по отношение разширяване на функционалностите;
- Приложима за разработката на всякакви по вид уеб сайтове;
- Лесно оптимизируема;
- Може да бъде персонализирана без предварително познаване на езици за програмиране.
- Подходяща е за средни и големи уеб сайтове;
- Висока степен на сигурност.

Критиците на системата дефинират нейните слаби страни така:[2]

- Joomla е трудна за потребители с ниска техническа подготовка;
- Само част от плъгините са безплатни;
- Трудности при SEO оптимизацията.

Фигура . представя *Back-end View* частта, през която се осъществява достъп до основните настройки и възможности за управление на уеб съдържанието. На Фигура. е представена *Joomla Front-end View* частта, представяща страниците на уеб сайта, така както изглеждат те след инсталацията на системата при наличие на готово съдържание (simple data).



Фигура 2.4. Back-end View на Joomla, версия 3.3



Фигура 2.5. Demo на Front-end View на Joomla, версия 3.3

### 2.4.3. CMS Drupal

CMS Drupal е система с отворен код, станала популярна сред много хора по света. Тази система обикновено се използва за изграждане на уеб сайтове, но може да бъде прилагана за създаване на търговски приложения и за управление на знанието. Drupal се разпространява под условията на GNU General Public License, което гарантира, че софтуера е свободен за изтегляне, управляване и споделяне.

Ето и някои от по-важните характеристики на системата:[2]

- Drupal е написана на PHP;
- Поддържа база данни MySQL;
- Съдържанието се създава с помощта на WYSIWYG редактор;
- Предоставя възможности за разширение на функционалностите.
- В Drupal малките програми, които добавят функции се наричат модули.

Основните предимства на CMS Drupal са:

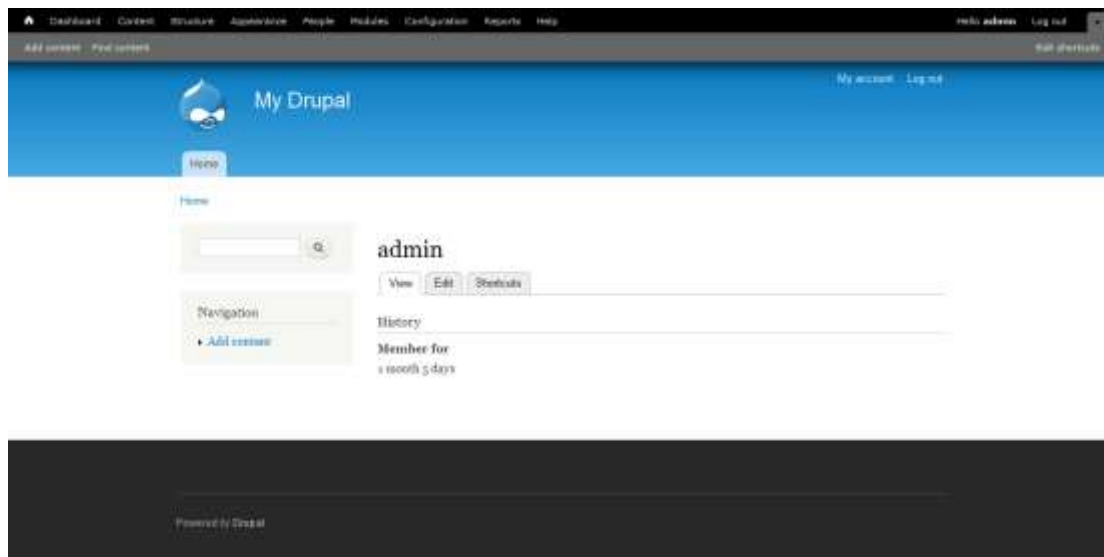


- Лесен за използване както от професионалисти, така и от непрофесионалисти;
  - Достъп до хиляди бесплатно разпространявани възможности за разширяване на уебсайта;
  - Поддръжка на темплейт система, която дава възможност на потребителите да посочат една измежду съществуващите теми;
  - Поддържа повече от 70 международни езици.
- Недостатъците на Drupal са изброени по-долу:[2]
- Трудности при обучение за работа със системата;
  - Текстовият редактор WYSIWYG е добавка, която не върви в пакета на системата;
  - Не е подходяща за малки сайтове;
  - Предоставят объркващ и неудобен за използване интерфейс на административната част.

На Фигура 2.2. е представен началния екран на системата Drupal, а на последващата фигура е представена страницата на административния панел при въведени параметри за достъп до административните функции.



**Фигура 2.2.** Начален екран на CMS Drupal при въведени параметри за вход (Демо версия)



Фигура 2.3. Административен панел на CMS Drupal при въведени параметри за достъп

#### 2.4.4. CMS Plone

Plone е система с отворен код, лицензиран под GNU General Public License (GPL), който позволява на всеки да използва източника безплатно. Това дава редица предимства като например: [2]

- Всеки потребител може да разгледа Plone кода и да го променя според своите нужди.
- Не изисква лицензионни такси;
- Притежава голяма потребителска база и легион от разработчици, експерти, преводачи, както и графични дизайнери, които развиват системата.

Plone поддържа инсталатори за Windows, Linux, и Mac. Всяка нова версия поддържа миграционни пътища и актуализации, така че всеки сайт, създаден с помощта на Plone да бъде „в крак с времето”. Няколко са характерните черти на Plone, които я правят предпочитано средство за създаване на уеб сайт от потребителите неспециалисти :[2]

- Потребителският интерфейс на Plone е преведен на повече от 35 езика. Езиковите пакети вървят пакетно със стандартната инсталация на системата.

- Plone осигурява високи нива на използваемост и достъпност. Той предоставя интерфейс, който е съобразен със стандарта WAI -AAA (Web Accessibility Initiative). Това позволява на обекти, изградени с Plone да бъдат

използвани от хора със зрителни увреждания. В допълнение, това осигурява по-добра индексация при търсене в Google.

- Plone разделя съдържанието (контента) от действителните шаблони, използвани за представяне на съдържанието, което позволява независимо управление на контентите от визията на сайта и обратно.

- Plone предоставя възможност за промяна облика на сайта посредством инсталация на теми. Голяма част от темите са безплатно разпространявани.

- Plone разполага със система за регистриране на потребители, предоставяща възможност за отдаване на различни привилегии за достъп за всеки потребител.

- Лесно проследим и управляем жизнен цикъл на съдържанието и достъпното работно пространство. За всеки елемент от съдържанието в един Plone сайт, могат да бъдат настроени права за достъп и начините на взаимодействие с него.

- Възможности за разширяемост и надграждане на функционалностите - отвореният код на Plone позволява лесно да бъде променен и конфигуриран почти всеки елемент на системата. Освен това са достъпни безброй пакети и инструменти за Plone, осигуряващи надграждане на функционалностите на сайта.

Според създателите си, Plone е идеална за използване от проектни групи, общности, уеб сайтове, за външни и вътрешни мрежи тъй като притежава предимствата:[2]

- Инсталира се с помощта на инсталатор и може да се пристъпи към нейната експлоатация веднага след приключване на инсталацията.

- Изключително лесна за употреба - добавянето, обновяването и поддръжката на съдържанието, визията и функционалностите изисква набор от умения, често добивани след кратко обучение.

- Системата е технологично независима, може да работи съвместно с релационни бази данни, системи с отворен код, както и лицензни системи и

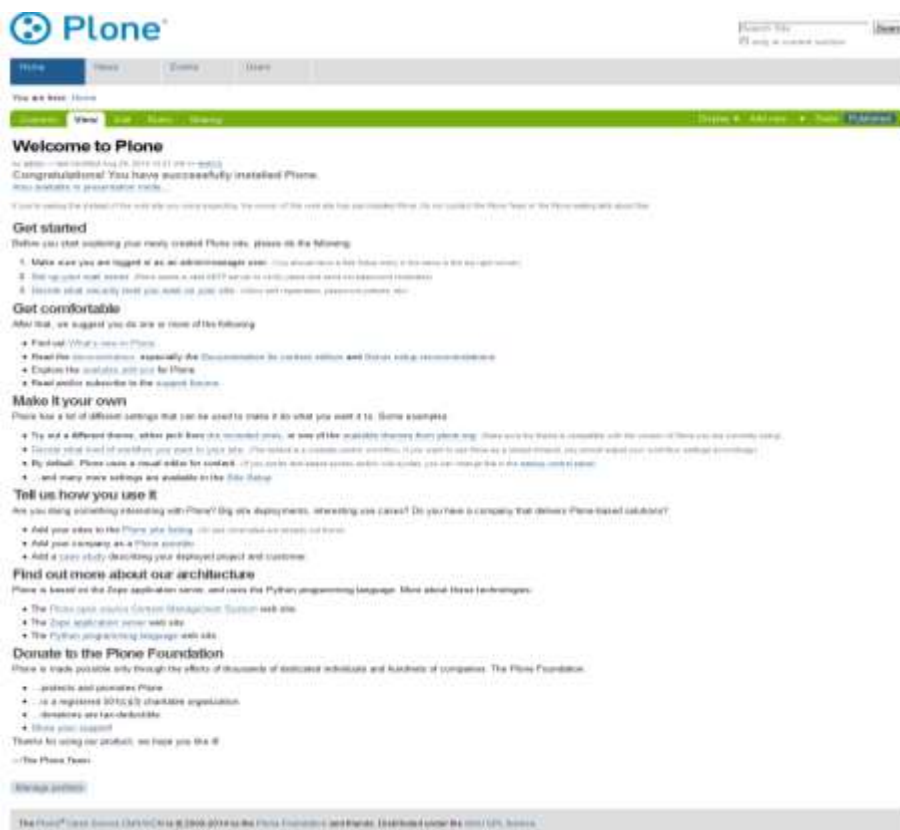
работи под голямо разнообразие от платформи, включително Linux, Windows, Mac OS X, Solaris и BSD.

Специфика на тази система е, че тя работи като приложение на Zope сървър приложения. Този сървър се инсталира автоматично от инсталатора по време на инсталационния процес. Основният език, използван в раждането на CMS Plone е Python. В последствие се включват JavaScript, XML и Perl.

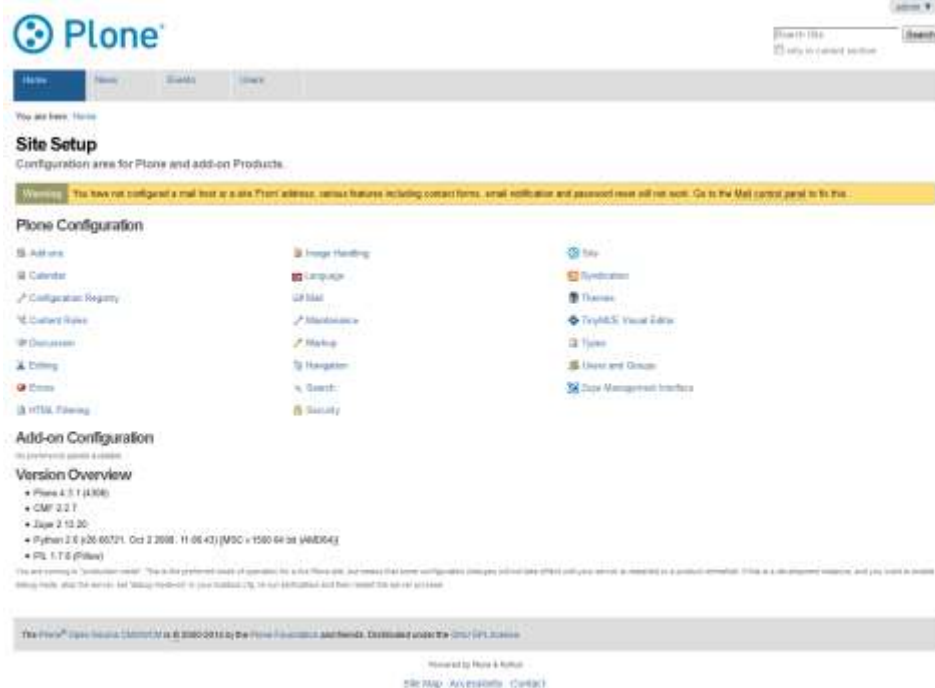
Въз основа на казаното дотук може да се обобщи, че основните предимства на Plone, които я правят предпочитана от потребителите са ползваемост, достъпност, гъвкавост и сигурност.

Недостатъците на Plone произхождат от бавната производителност на системата (често зависеща от използвания браузър). Съществува и проблем с кеширането – често при извършване на актуализации в сайта те не се виждат веднага поради поддръжката на кеш.

На Фигура 2.4. е представен началния екран на Plone, като изглежда системата веднага след нейната инсталация, а на Фигура 2.5 е представена страницата за управление на системата при въведени привилегии за достъп до настройките на сайта.



Фигура 2.4. Начален екран на системата Plone при въведени параметри за достъп



Фигура 2.5. Страница за управление на CMS Plone при въведени параметри за административен достъп

## 2.4.5. Сравнителен анализ

В следващата таблица е направено сравнение на разгледаните в раздела системи, базирано на някои общи характеристики, предимства и недостатъци на всяка от тях.[2]

Таблица 2.5. Сравнителна таблица

КМС	Удобства при употреба	Предимства	Недостатъци	База данни
<b>Drupal</b>	Предоставя WYSIWYG редактор, функция за управление на потребителите и широка подкрепа от Drupal общността.	Позволява на множество потребители да създават, редактират и публикуват съдържание.	Трудности при самообучение.	MySQL
		Интегрира се с различните социални медии (напр. Facebook).	Потребителският интерфейс е труден за възприемане от новите потребители.	
		Поддържа повече от 70 световни езици.	Не се препоръчва употребата на системата за малки сайтове.	
			Текстовият редактор	

		Drupal общност, Drupal форуми и дискуссионни групи.	WYSIWYG редактор не върви с инсталационния пакет.		
		Повече от 6000 безплатни добавки.	Трудности при инсталация и администриране.		
		Добър интерфейс за приложно програмиране.	Значителни разлики между отделните версии.		
<b>Joomla</b>	Мощен CMS, който е лесен за инсталиране и използване.	Изключително гъвкава.	Проблеми с производителността на системата.	MySQL	
		Лесно оптимизиране	Не се препоръчва за разработка на малки сайтове.		
		Поддържа многоезичност.			
		Подкрепа от Joomla общността.	Ограничен брой шаблони		
	Лесна за персонализация, не изискваща специфични технически познания.	Отдалечен достъп и контрол посредством уеб браузър.	Трудности в самообучението при първа среща със системата.		
		Лесен за поддръжка, дори и за начинаещите потребители.	Могат да бъдат изградени различни навигационни нива, които могат да доведат до грешки.		
<b>Plone</b>	Лесен за инсталиране и употреба.	Безплатна система с отворен код.	Бавна производителност на системата .	ZODB или MySQL	
		Високо ниво на сигурност.	Проблеми свързани с кеширането.		
		Достъп до широк спектър от добавки и плъгини.	По-малко функции за управление на БД.		
		Интеграция с други системи и			

		приложения.		
		Лесна за оптимизиране.	за	
<b>Word Press</b>	Лесна за използване и поддръжка. Идеален за потребители, които не притежават специални технически умения.	Възможност за разширяване на функционалността	за на	Поддръжка на постоянни актуализации.
		чрез специално проектирани плъгини.		Проблеми с производителността – инсталацията на много добавки може да забави един сайт, създаден с WordPress.
		Удобна за осъществяване на управление на един или няколко блога.		Уязвим за хакери.
		Поддръжка на SEO инструменти за оптимизация		Трудности в самообучението при първа среща със системата.
		Подкрепа от WordPress общността.		
		Лесна за ъпгрейд.		Все още не е пълна КМС.
		Достъп до Codex: урок уебсайт.		

Въз основа на направеното сравнение може да бъде изведено основното обобщение, че предимствата осигурявани от разгледаните системи са аналогични и задоволяват потребностите на детското заведение. От своя страна обаче авторката на настоящата дипломна работа счита, че най-подходяща за разработката на проекта е CMS Joomla поради факта, че тя е средство, използвано от непрофесионалисти както за малки, така и за по-големи сайтове. Освен това системата предоставя редица безплатни възможности за надграждане, както и достъп до богата документация.

## РЕЗУЛТАТИ И ИЗВОДИ ОТ ВТОРА ГЛАВА

При избор на контент мениджмънт система, основната цел е да се намери такава която успешно да балансира между дизайн, използваемост и разширяемост на функционалностите. Съществува голямо разнообразие от КМС, измежду които трябва да бъде посочена тази, която в най-голяма степен отговаря на целите за направа на дадения уеб сайт.

След направения анализ на видовете КМС и направения извод, че за нуждите на настоящата дипломна работа трябва да бъде избрана система за управление на съдържание, отговаряща на критериите “отворен код” и „управляваща уеб контент” за направата на уеб сайт за детска градина бяха разгледани по- подробно видовете WCMS с отворен код.

Въз основа на изнесената в главата информация може да се обобщи, че всяка от разгледаните WCMS с отворен код предлагат еднотипни възможности и функционалности и при сравнението на тези системи се откриват повече сходства, отколкото различия. Но все пак, след направеното сравнение на характеристиките, предимствата и недостатъците на някои от по-популярните WCMS с отворен код, може да се каже, че Joomla притежава много силни страни, но и притежава голяма лекота на използване. Тя е лесна за инсталация, употреба и поддръжка, с висока степен на гъвкавост по отношение разширяване на функционалностите и висока степен на сигурност. Joomla е система, която поддържа всички изисквания за успешно създаване на сайт за детска градина.



## **ГЛАВА ТРЕТА: ИЗГРАЖДАНЕ НА УЕБ САЙТ ЗА ДЕТСКА ГРАДИНА №182 „ПЧЕЛИЦА“**

Процесът по създаването на уеб сайт, независимо от това какъв е неговия вид ( новинарски, корпоративен, развлекателен, онлайн магазин или друг) или целева група към която е насочен, може да се сведе до няколко етапа. Основните етапи в разработката на всеки уеб сайт са:

- Определяне на целите на сайта
- Уточняване на съдържанието
- Изработване на сайта
- Публикуване на сайта
- Поддържане и актуализиране

В настоящата глава ще бъде разгледан всеки един от тези етапи.

### **3.1. Дефиниране на изискванията към сайта и аспектите на информационната му архитектура**

Уеб сайтове се правят поради различни причини и цели, но все пак едно от основните предназначения на всеки уеб сайт е да информира. Същевременно колкото по-атраکتивен е той, толкова по-голяма популярност ще придобие. От една страна, той трябва да е с добър външен вид, от друга – да е полезен и удобен за употреба. В крайна сметка може да се обобщи, че основните принципи при създаването на всеки уеб сайт са атрактивност, полза и достъпност.

Основната идея и цел на настоящата дипломна работа е сайтът, който ще бъде изграден да информира родителите на децата записани в детска градина №182 „Пчелица“ за всичко свързано с обучението и ежедневието на децата в детската градина и заедно с това да бъде цветен и забавен, за да се хареса и на децата.

За да може да бъде уточнено съдържанието на уеб сайта първо трябва да бъдат поставени както основните изисквания към всеки сайт на детска градина така и специфичните изисквания от страна на детската градина.

### **3.1.1. Изграждане на списък с изисквания към уебсайта, базиран на анализ на конкурентни сайтове**

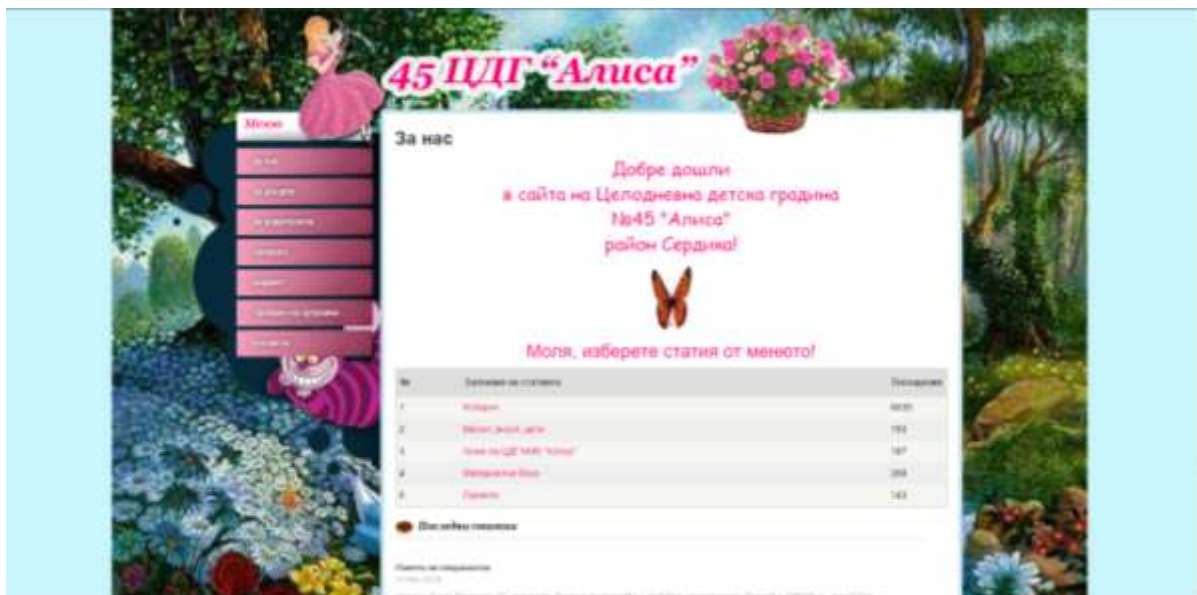
За да се установят основните изисквания към всеки сайт на детска градина трябва да бъде направен анализ на съществуващи уеб сайтове на детски градини. Анализът ще бъде направен върху дизайна, функционалностите и информацията предоставена на потребителите на 5 случайно избрани сайтове на детски градини, които са посочени в следващата таблица.

Таблица 3.1. Представителна извадка на сайтове на детски градини

<b>№</b>	<b>Име на детска градина</b>	<b>URL адрес</b>
1.	45 ЦДГ „Алиса“	<a href="http://cdq45alisa.com/">http://cdq45alisa.com/</a>
2.	ЧДГ „Къщата на Алиса и нейните приятели“	<a href="http://alishahouse.org/">http://alishahouse.org/</a>
3.	ЧДГ „Бръмбазък“	<a href="http://www.brambazak.bg">http://www.brambazak.bg</a>
4.	ЦДГ 167 „Малкият принц“	<a href="http://malkiyatprinc.eu/wordpress/">http://malkiyatprinc.eu/wordpress/</a>
5.	ЧДГ „Патета“	<a href="http://www.pateta.bg/bg.html">http://www.pateta.bg/bg.html</a>

#### **3.1.1.1. Анализ на дизайна на сайтовете**

На Фигура 3.1. е показан сайтът на Целодневна Детска Градина № 45 „Алиса“. Дизайнерът изцяло се е съобразил с целевата аудитория (малките деца и родителите им). Дизайнът на сайта е цветен и забавен. В средата е изчистен и опростен, което поддържа вниманието на посетителите върху дадената информация. С тази визия сайтът ще се харесва на децата, но също така и правилно ще извършва основната си цел – да информира родителите. Поставена е вертикална навигация, която е позиционирана в лявата част на сайта, което подпомага за добрия баланс между елементите в сайта. Няма ненужни допълнителни елементи освен броят посещения на дадена статия, което за този тип сайт няма значение и затова не би трябвало да бъде показвано.



Фиг. 3.1. Начална страница на 45 ЦДГ „Алиса“

На Фигура 3.2. е представен сайта на ЧДГ „Къщата на Алиса и нейните приятели“. Тук отново за фон е избрана весела и забавна картинка, но при нея основната рисунка е от лявата част на екрана, а дясната е прекалено изчистена, което привлича погледа от лявата част на сайта, а не върху предоставената информация. Също така има повторение в менютата, което е ненужно и единствено обърква потребителите. Изборът на слайдшоу поставено на началната страница е добър избор. Чрез него и кратката информация дадена към всеки елемент от слайдшоуто се подканват потребителите да разгледат други части на сайта, което може да не стане, ако информацията бъде поднесена по друг начин.



Фиг. 3.2. Начална страница на ЧДГ „Къщата на Алиса и нейните приятели“

На Фигура 3.3. е показан сайтът на Частна Детска Градина „Бръмбазък“. В този сайт дизайнерското решение за фон отново е ярка, цветна картинка. Избраното цветово оформление подхожда за използване в сайт за детска градина, но чрез подбора на толкова много цветове и големи обекти вниманието на посетителя на сайта се отвлеча ненужно. Избрано е меню с икони, което отговаря за целевата група, но при този избор на тези ярки цветове и големината на другите елементи се получава претрупване.



Фиг. 3.3. Начална страница на ЧДГ „Бръмбазък“

На Фигура 3.4. е показан сайтът на ЦДГ 167 „Малкият принц“. При този сайт дизайнът е изчистен и семпъл, но липсва съразмерност между менюто и останалите елементи. В горната част се вижда струпване на елементи. Логото и полето за търсене не са на правилните позиции. Също така там се вижда линк „история“, които не би трябвало да бъде поставен на тази позиция. Същият линк се вижда малко по-надолу на страницата, което довежда до мисълта, че горният не би трябвало въобще да съществува. Заради тези недостатъци сайтът изглежда недовършен.



Фиг.3.4. Начална страница на ЦДГ 167 „Малкият принц“

На фиг. 3.5. е показан сайтът на ЧДГ „Патета“. Прави впечатление добрата визия на сайта и конкретно насочената информация към различните целеви групи (навигационното меню в горната част на сайта и навигационното меню в долната част). Въпреки многото цветове и елементи се наблюдава добре балансиран и приятен външен вид. „Стаите“ в къщичката също са навигационно меню. Изборът на този дизайн е забавен и подканва потребителите да разгледат целия сайт.



Фиг. 3.5. Начална страница на ЧДГ „Патета“

След направения анализ на дизайна на петте случайно избрани сайта, може да се заключи, че визията на сайтовете трябва да бъде пъстра, цветна и забавна, но да се внимава да не се претрупа, за да не обърка или излишно да отвлича вниманието на потребителите. Трябва да е съобразен както с това да се харесва на децата, но и с това да предостави нужната информация на потребителя без излишно разсейване от страна на цветовете и елементите избрани за сайта.

### 3.1.1.2. Анализ на функционалностите в сайтовете

В този раздел ще бъдат разгледани какви функционалности се поддържат от примерните сайтове и дали има такива които задължително трябва да присъстват в един сайт на детска градина.

Таблица 3.2. - Анализ на функционалностите в примерните сайтове

№	Име на детска градина	URL адрес	Регистрацион на форма	Карта	Галерия	Календар	Часовник
1.	45 ЦДГ „Алиса“	<a href="http://cdq45alisa.com/">http://cdq45alisa.com/</a>	Не	Да	Да	Не	Не
2.	ЧДГ „Къщата на Алиса и нейните приятели“	<a href="http://alisahouse.org/">http://alisahouse.org/</a>	Не	Да	Не	Не	Не
3.	ЧДГ „Бръмбазък“	<a href="http://www.brambazak.bg">http://www.brambazak.bg</a>	Не	Да	Да	Не	Не
4.	ЦДГ 167 „Малкият принц“	<a href="http://malkiyatprinc.eu/wordpress/">http://malkiyatprinc.eu/wordpress/</a>	Не	Не	Да	Не	Не
5.	ЧДГ „Патета“	<a href="http://www.pateta.bg/bg.html">http://www.pateta.bg/bg.html</a>	Не	Не	Да	Не	Не

Обобщението което може да бъде направено след анализа на функционалностите в примерните сайтове е: Няма задължителни функционалности, които трябва да се съдържат в един сайт на детска градина, за разлика от училищния уеб сайт, в които задължително трябва да има електронен дневник. Избора дали да има и какви да бъдат функционалностите се прави изцяло от определената детска градина в зависимост от целите поради които се създава сайта.

### **3.1.1.3. Анализ на информацията предоставена в сайтовете**

За да бъде полезен уеб сайтът трябва да има добре структурирана информация насочена към целевата група потребители. Тя трябва да бъде ясна, точна и актуална. Информацията, предоставена в един сайт на детска градина обикновено включва:

- Начална страница с кратко представяне на градината
- Страница с разпределението на децата в групите
- Страница с представянето на екипа работещ в детското заведение
- Страница с дневния режим
- Страница със съобщения и новини
- Страница с допълнителните дейности предоставени от детската градина
- Страница съдържаща снимки (галерия)
- Страница с контакти
- Страница с обществените поръчки на детското заведение

### **3.1.2. Изисквания към уеб сайта от страна на детската градина**

Всеки човек или организация има своя гледна точка кое е важно да бъде показано в техния сайт и каква визия да има той. Същото се отнася и за гледната точка детска градина. Поради тази причина, в създаването на настоящият сайт е изключително важно да се вземат предвид изискванията към уеб сайта предоставени от страна на детската градина:

- Страница със седмично меню

- Страници с начислените такси, които да са достъпни само за регистрирани потребители
- Дизайнът да съдържа елементи свързани с името на детската градина ( картинки с пчелички, повечето от които да бъдат анимирани).

След като беше уточнен дизайна, функционалностите и съдържанието на уеб сайта спрямо задължителните елементи, които трябва да съдържа всеки сайт на детска градина и спрямо изискванията поставени от страна на самата детска градина, следващият етап в разработката му е неговото създаване. Изработването на сайта ще бъде разгледано в следващият раздел.

### **3.2. Етапи от изработката на уебсайта**

Изхождайки от изискванията и задължителните елементи които трябва да присъстват в един сайт на детска градина, както и от направения анализ на примерните уеб сайтове разгледани в предишната глава може да се изведат основните стъпки, чрез които ще бъде създаден настоящият сайт. Стъпките по които ще бъде създаван сайта са:

- Избор и инсталация на темплейт
- Избор и изграждане на навигационно меню
- Създаване на начална страница и вътрешни страници
- Добавяне на „Карта”
- Създаване на „Галерия”
- Създаване на регистрационна форма и задаване на права на потребителите от администрацията на детското заведение

В следващите раздели ще бъдат разгледани всяка една от тях.

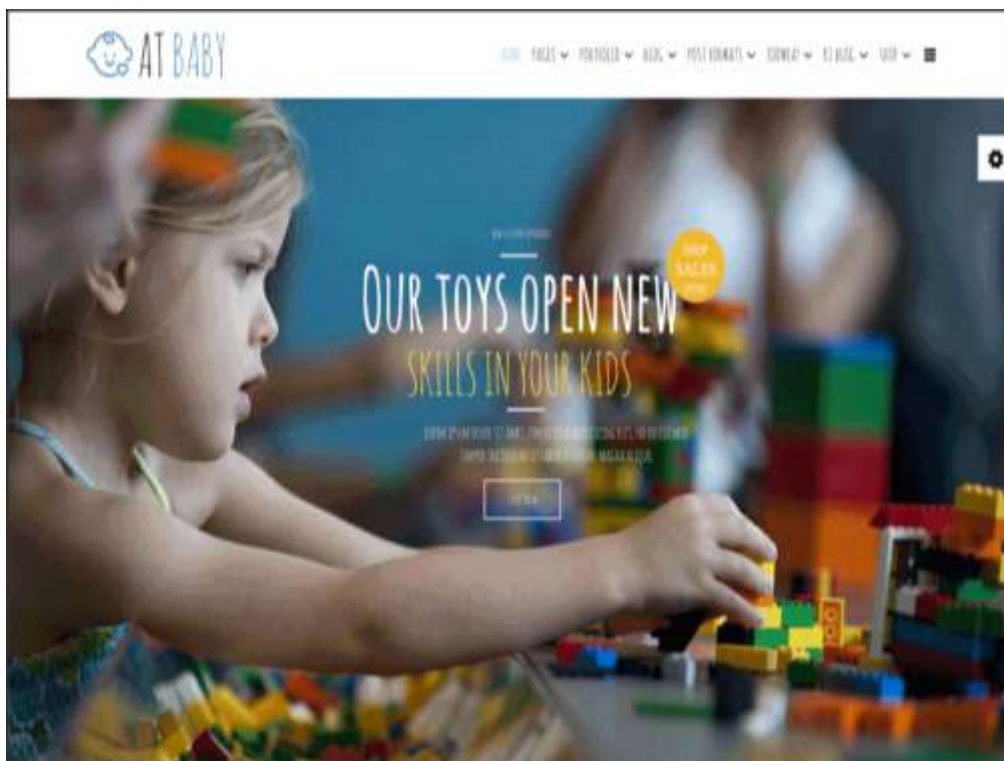
#### **3.2.1. Избор и инсталация на темплейт**

При избор на темплейт е необходимо съобразяването с изискването сайта да се харесва на децата, да бъде цветен и забавен. Поради тази причина е избран темплейт „at\_baby” от адреса <http://templatki.com/template/915/at-baby-joomla>.



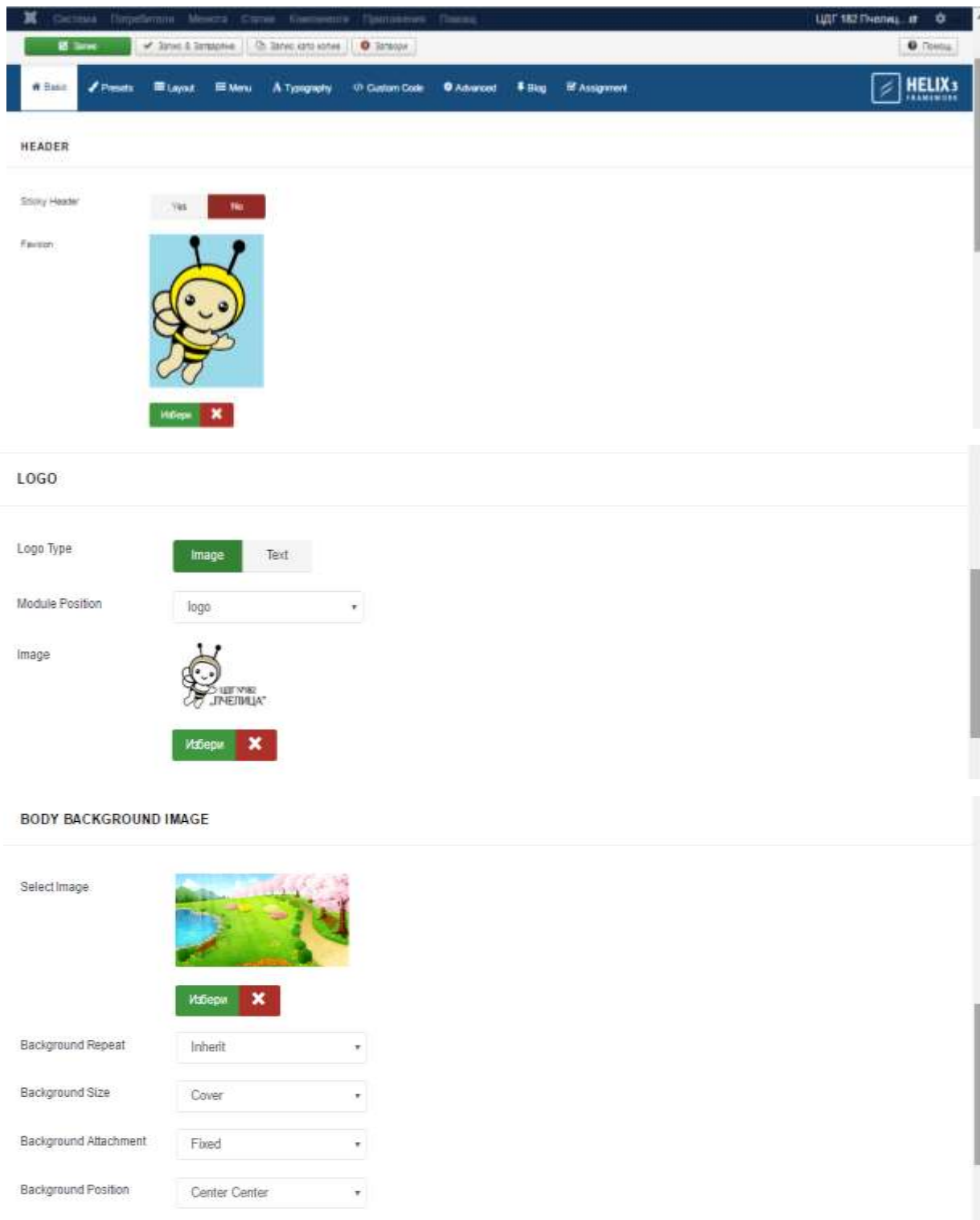
Темплейта „at\_baby” е безплатен Joomla шаблон, който е насочен за деца / детски сайтове . Този шаблон поддържа мобилна / таблетна версия и 4 цветни стила. Шаблонът използва най-новите Bootstrap CSS рамки. Също така е напълно отзивчив (Fully responsive). Поддържа Off-Canvas Menu и Mega Menu; Optional Fixed (sticky) Header; Desktop, Mobile and Retina лого опции и много други.

На следващата фигура е показана началната страница на темплейта.



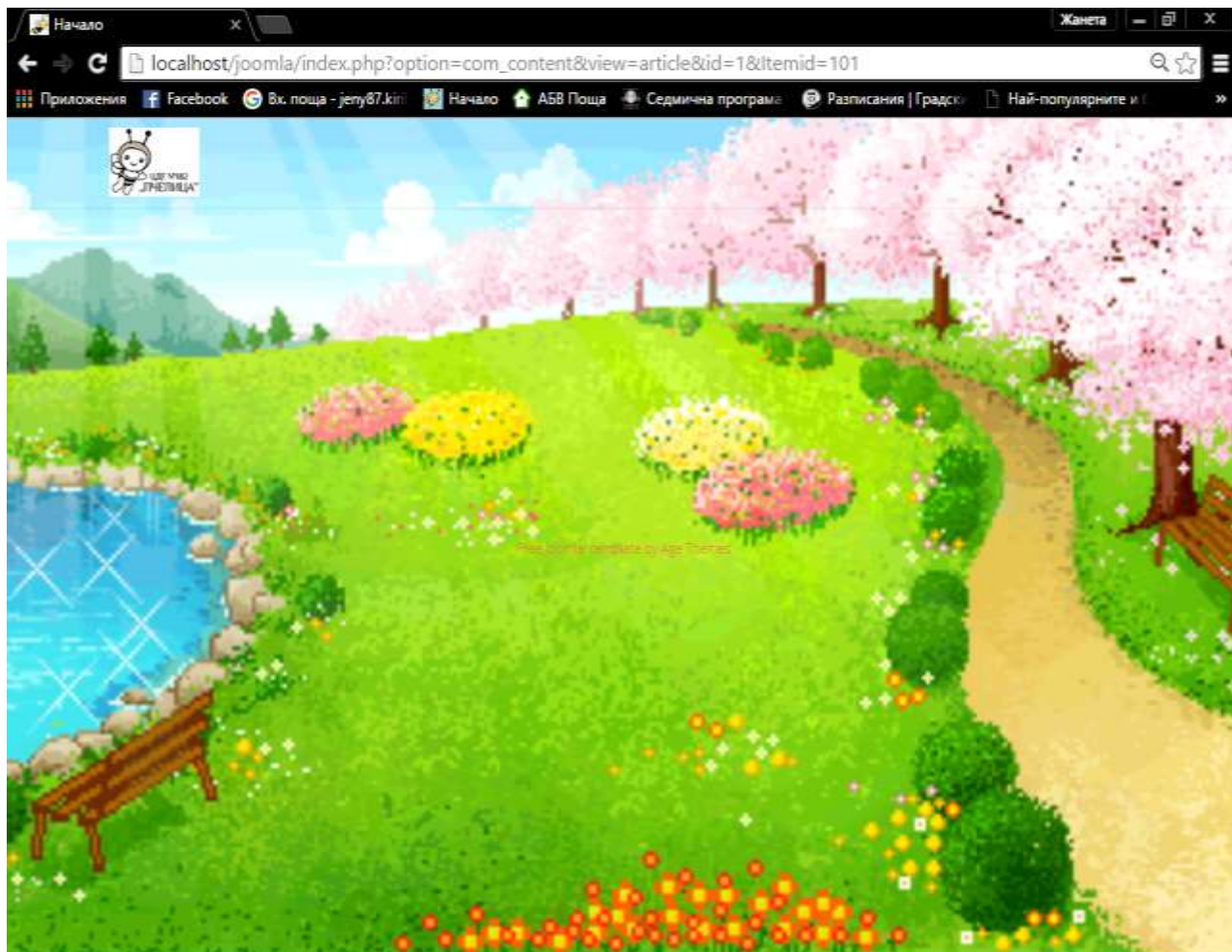
Фиг. 3.5. – Начален екран на темплейт at\_baby

За настоящият сайт за фон на темплейта е избрана картинка отговаряща на изискванията поставени в предишния раздел – да е цветна, забавна, анимирана и със връзка с наименованието на детската градина . За лого и фавикона са поставени такива предоставени от детското заведение. Фонът, както и логото и фавиконата за сайта се добавят от административния панел чрез редакция на шаблона. Страницата за редакция на темплейта е показана на следващата фигура.



Фиг. 3.6.- Страница за редакция на темплейт at\_baby

Визията на сайта след инсталацията на избрания темплейт и след добавяне на фона, логото и фавиконата е показан на следващата фигура.



Фиг. 3.7. - Начална страница след инсталация на темплейт

### 3.2.2. Избор и изграждане на навигационно меню

Навигационното меню е една от най-важната част от направата на един уеб сайт. Чрез него се достига до всички елементи (страници) на сайта. То може да бъде хоризонтално или вертикално в зависимост от това как искаме да изглежда сайта. Също така може едновременно в един сайт да има както хоризонтално така и вертикално меню.

Поради важността на едно навигационно меню за всеки един сайт, в настоящата дипломна работа ще бъде разгледано малко по-подробно избора и изграждането му.

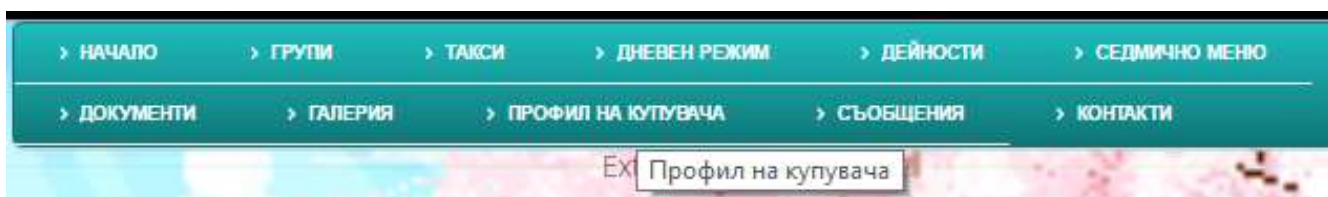
Бутоните който ще включва менюто, какъв тип бутон е всеки от тях, както и кратки описания към всеки един от тях са предоставени в Таблица 3.3.

Таблица 3.3. - Описание на навигационното меню

Име на бутон	Тип на бутон в менюто	Описание
<i>Начало</i>	Основен запис	Съдържа кратко представяне на детската градина
<i>Нашият химн</i>	Подменю на „Начало”	Съдържа химна на детската градина
<i>Нашият екип</i>	Подменю на „Начало”	Представя екипа работещ в детското заведение
<i>Групи</i>	Основен запис	Съдържа кратко описание на разпределението на групите и линкове към всяка група
<i>Дъга</i>	Подменю на „Групи”	Съдържа списък на децата от група „Дъга”
<i>Слънце</i>	Подменю на „Групи”	Съдържа списък на децата от група „Слънце”
<i>Такси</i>	Основен запис	Съдържа линкове към страници с всички такси за определен месец и линк към архива с такси
<i>Детска градина</i>	Подменю на „Такси”	Съдържа линкове към страници с всички такси за детска градина за цялата година
<i>Народни танци</i>	Подменю на „Такси”	Съдържа линкове към страници с всички такси за народни танци за цялата година
<i>Приложни изкуства</i>	Подменю на „Такси”	Съдържа линкове към страници с всички такси за приложни изкуства за цялата година
<i>Английски език</i>	Подменю на „Такси”	Съдържа линкове към страници с всички такси за английски език за цялата година
<i>Архив</i>	Подменю на „Такси”	Съдържа линкове към страници с всички такси за цялата година
<i>Дневен режим</i>	Основен запис	Съдържа описание на дневния режим на децата

<b>Дейности</b>	Основен запис	Съдържа кратки описания на дейностите, който се предлагат в детската градина
<b>Английски език</b>	Подменю на „Дейности”	Съдържа снимки от откритите уроци по Английски език
<b>Народни танци</b>	Подменю на „Дейности”	Съдържа снимки от откритите уроци по Народни танци
<b>Приложни изкуства</b>	Подменю на „Дейности”	Съдържа снимки от откритите уроци по Приложни изкуства
<b>Седмично меню</b>	Основен запис	Съдържа седмичното меню е детското заведение
<b>Документи</b>	Основен запис	Съдържа PDF файлове на документи свързани с дейността на градината, както и линкове за изтеглянето им docx формат
<b>Профил на купувача</b>	Основен запис	Съдържа PDF файлове на документи свързани с обществените поръчки на детската градината
<b>Галерия</b>	Основен запис	Съдържа снимки представящи материалната база в градината
<b>Съобщения</b>	Основен запис	Съдържа важни съобщения до родителите, както и съобщения за предстоящи събития
<b>Контакти</b>	Основен запис	Съдържа адрес и карта за местоположението на детската градина, както и работното време на градината и персонала. Също така има списък за контакт (email адреси и телефони) и линк към страницата на заведението във Facebook

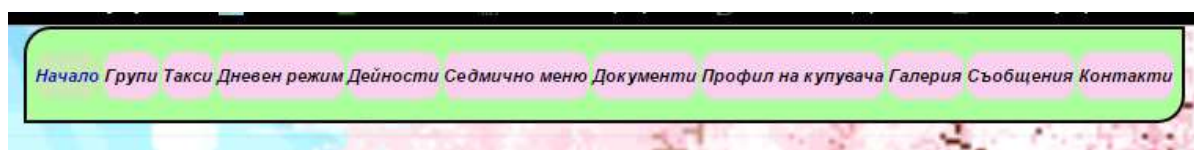
За стил на менюто бяха предложени три варианта, един от който да бъде избран от администрацията на детската градина за направата на настоящият сайт. Трите вида менюта са показани в следващите три фигури.



Фиг. 3.4 – Навигационно меню Вариант 1

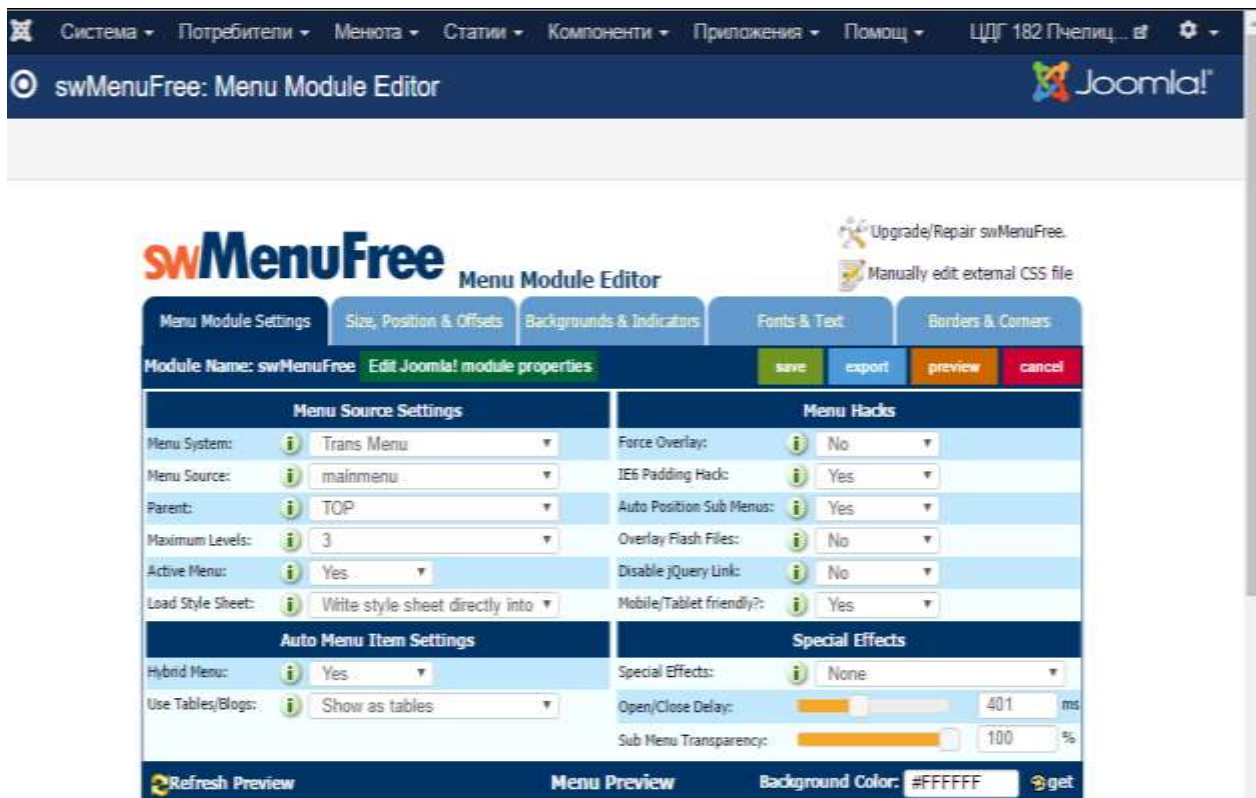


Фиг. 3.5. – Навигационно меню Вариант 2



Фиг. 3.6. – Навигационно меню Вариант 3

Навигационно меню Вариант 1 е със стилен и изчистен вид, но не поддържа голям избор за редакция на цветовете, шрифт и големина и разредка между елементите му. Навигационно меню Вариант 2 е меню с икони. Това отговаря на целевата група, но при направения избор за темплейт и желанието на администрацията в сайта да присъстват множество анимации с добавянето на подобно меню визията на сайта става претрупана. Поради тези причини меню Вариант 1 и меню Вариант 2 бяха отхвърлени. Избрано беше меню Вариант 3 поради фактът, че модулът съдържа голям набор от свойства, които могат да бъдат редактирани. Част от редакционните свойства на модула са показани в следващата фигура.



Фиг. 3.7. – Страница за редакция на свойствата на менюто

### 3.2.3. Създаване на начална страница и вътрешни страници

При създаването на страниците трябва да се обърне внимание както на това как изглеждат повечето сайтове на детски градини, така и да бъде съобразена визията с желанията на детското заведение.

Беше предоставен избор за дизайна на статичните страници в сайта (Начало, Нашият химн, Нашият екип, Групи, Дъга, Слънце, Дневен режим, Съобщения) на администрацията на детското заведение. Предложение № 1 е текстът да бъде написан в изчистена, едноцветна правоъгълна форма с фон подобен на фона на темплейта. Предложение № 2 е текстът да бъде написан върху картинка с пчелички, за да отговаря на изискването да има елементи свързани с името на детската градина. Изборът на администрацията беше текстът да бъде написан върху картинка отговаряща на изискването темата на картинките да бъде свързана с името на детската градина. Двата варианта на дизайн на една от статичните страници в сайта са показани в следващите две фигури.



Фиг. 3.8. – Вариант на дизайн на статична страница с таблица



Фиг. 3.8. – Вариант на дизайн на статична страница с текст върху картинка

За динамичните страници (Седмично меню, Такси, Детска градина, Народни танци, Приложни изкуства, Английски език, Дейности), беше избран вариантът с изчистена, едноцветна правоъгълна форма или таблица с фон



подобен на фона на темплейта, за да може лесно да бъдат нанасяни редакции на текста и елементите. Във Фигура 3.9 е показана как изглежда една от динамичните страници.

Дата: 08.11.2014 - 12.11.2014	Понеделник	Вторник	Сряда	Четвъртък	Петък
<b>Закуска</b>	Кускус с зехте и сирене	Сандвич с масло и колбас Чай с лимон	Тутмания Прясно мляко	Сандвич с масло и варено мляк Чай с лимон	Сандвич с масло и извара Чай с лимон
<b>10 часа</b>	Плод - Ябълка	Плод - Грозде	Плод - Грозде	Плод - Грозде	Плод - Грозде
<b>Обяд</b>	Супа топчета Домати с ориз на фурна Ябълка	Супа със спанак Свинско със зеле Банан	Супа борш Месо с грав Банан	Зеленичукова супа Пълнозърниста салата Ябълка	Супа леща Риба на фурна Ябълка
<b>Следобедни закуски</b>	Месо с кисело мляко	Млечна баница	Фюли с масло и лютеница Айран	Карофлейкс с прясно мляко	Бутерка с лютеница Айран

Фиг. 3.9. – Изглед на динамична страница

За дизайна на Таксите първоначално беше избрано имената на децата в групите да бъдат линкове препрращащи към страница с всички такси за определено дете и крайна обща сума. Поради фактът, че в детската градина са записани над 50 деца и за да бъдат нанасяни таксите на всяко по отделно трябва да се редактират множество страници, след едномесечно ползване този дизайн беше отхвърлен и таксите започнаха да се нанасят в 4 страници за всяка дейност подлежаща на таксуване за всички деца едновременно създадена по шаблон предоставен от детската градина и линковете към имената на децата бяха премахнати. В следващите 4 фигури са представени съответно първоначалния избор за дизайн с линкове на една от групите, една от страниците с обща такса за съответно дете, дизайн на страница без линкове на една от групите и една от страниците с общи такси за съответна дейност.



Фиг. 3.10. - Дизайн на страница Група Слънце с линкове

Начало Група Такса Дневен режим Дейности Социално мени Документи Газария Профит на музика Свободна Компания

Месец ОКТОМВРИ 2015г.	Такса	Платено
Такса детска градина	10 лв.	Да
Такса вода, картон, чашки	5 лв.	Да
Такса народни танци	20 лв.	Да
Такса английски език	25 лв.	Да
Такса приложни изкуства	22 лв.	Да
Общо:	102 лв.	

Фиг. 3.11. – Примерна страница с обща такса на едно от децата



**Група "СЛЪНЦЕ"**

Александра Сидова	Ирена Градина
Александър Михайлов	Кристина Христова
Антонина Ризикова	Кристина Цветков
Борис Николаев	Кристина Ризикова
Борис Ризик	Лора Иванова
Валентина Григорова	Надя Сметаларова
Васил Андреев	Надя Петрова
Велин Андреев	Нелица Кръстева
Венери Божикова	Николай Недялков
Виктория Борисова	Раица Младенова
Гергана Благорова	Санджа Кръстева
Гергана Вельова	Симона Вълчева
Евгени Ризиков	Юлия Симеонова
Илия Илия	Миа Сидова

Фиг. 3.12. - Дизайн на страница Група Слънце без линкове

**ТАКСИ**

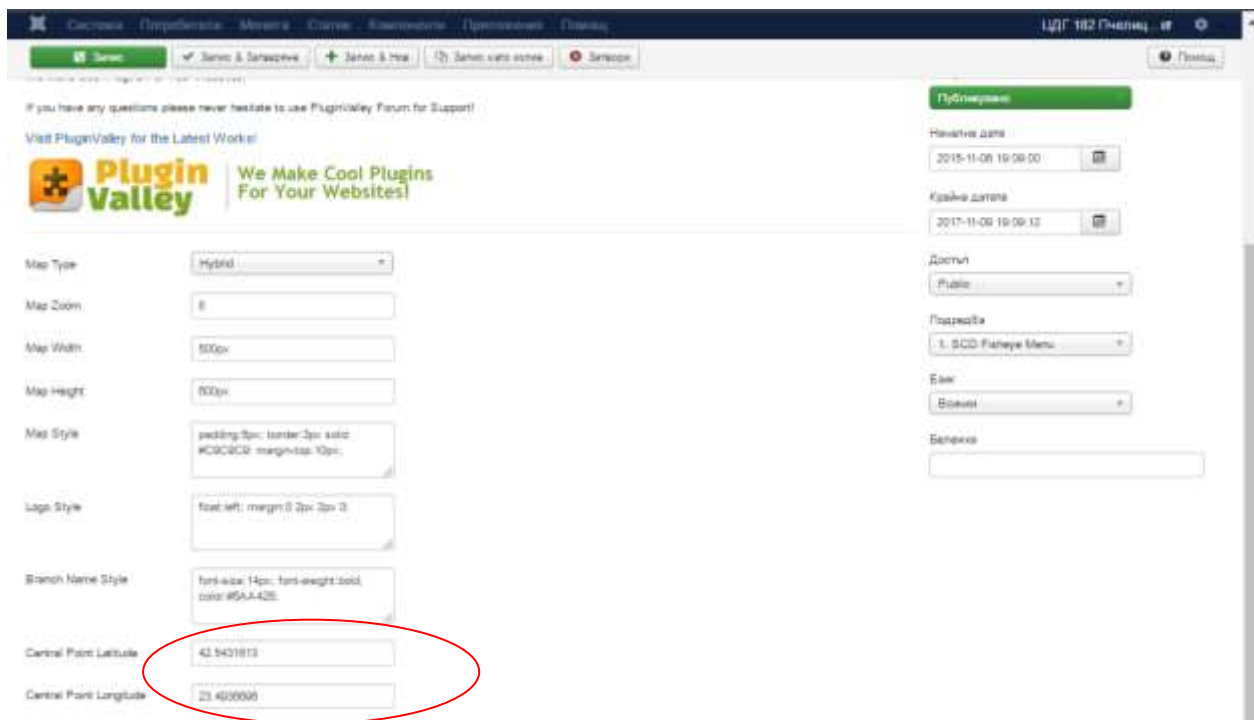
ТАКСА за месец ЯНУАРИ 2016г. на Група „СЛЪНЦЕ“					ТАКСА за месец ЯНУАРИ 2016г. на Група „ОБЛА“				
Име, Фамилия	Месечна такса	Дневна такса	Изчислена такса	Уплатено	Име, Фамилия	Месечна такса	Дневна такса	Изчислена такса	Уплатено
Александра Сидова	30лв.	1,43лв.	12,87лв.	да	Александър Михайлов	60лв.	2,86лв.	48,62лв.	да
Александър Михайлов	60лв.	2,86лв.	31,46лв.	да	Виктор Свистов	60лв.	2,86лв.	31,46лв.	да
Антонина Ризикова	60лв.	2,86лв.	48,62лв.	да	Виктор Христова	60лв.	2,86лв.	48,62лв.	да
Борис Николаев	60лв.	2,86лв.	45,76лв.	да	Виктор Христова	60лв.	2,86лв.	45,76лв.	да
Борис Ризик	60лв.	2,86лв.	48,62лв.	да	Виктор Христова	60лв.	2,86лв.	37,18лв.	
Валентина Григорова	60лв.	2,86лв.	45,76лв.	да	Виктор Христова	60лв.	2,86лв.	40,04лв.	да
Васил Андреев	60лв.	2,86лв.	45,76лв.	да	Виктор Христова	60лв.	2,86лв.	40,04лв.	да
Велин Андреев	60лв.	2,86лв.	40,04лв.	да	Виктор Христова	60лв.	2,86лв.	37,18лв.	да
Венери Божикова	30лв.	1,43лв.	24,31лв.	да	Илия Андреев	30лв.	1,43лв.	20,02лв.	да
Виктория Борисова	60лв.	2,86лв.	45,76лв.	да	Илия Сметаларова	60лв.	2,86лв.	20,02лв.	да
Гергана Благорова	60лв.	2,86лв.	25,74лв.	да	Илия Сметаларова	60лв.	2,86лв.	34,32лв.	да

Фиг. 3.13. – Дизайн на страница Такси Детска градина

### 3.3. Добавяне на основните функционалности в сайта

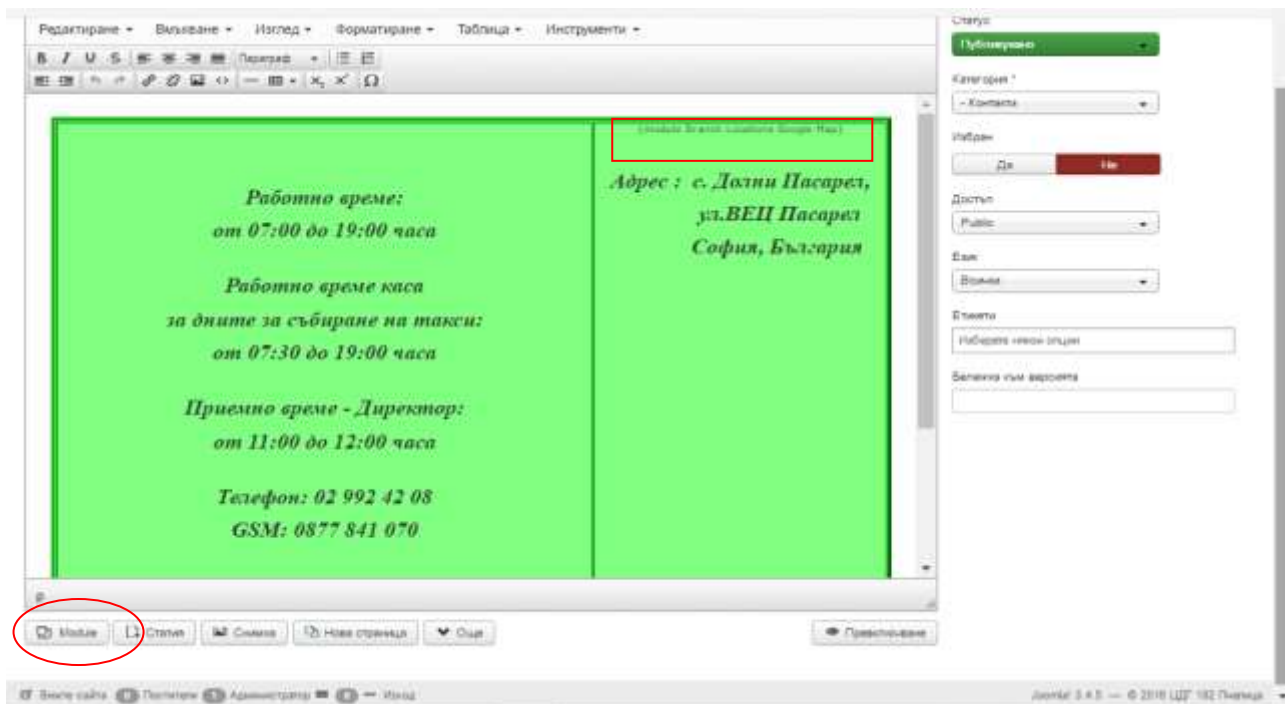
#### 3.3.1. Добавяне на карта „Карта”

За добавяне на Карта беше изтеглен модулът Branch Locations Google Map от сайта <http://extensions.joomla.org/extension/multiple-locations-google-map>. В страницата за редакция на модула бяха зададени координатите на местонахождението на детската градина, за да може картата да се позиционира на правилното място. Това е показано на следващата фигура.



Фиг. 3.14 – Страница за редакция на модул Branch Locations Google Map

За да бъде добавен модулът към статия „Контакти” беше изтеглена добавката „Button - NoNumber Modules Anywhere” от сайта <http://extensions.joomla.org/extension/modules-anywhere>. Тази добавка позволява да се покаже бутон „Модул” в страницата за редакция на статии и чрез него да бъдат поставяни модули в статиите, а не в странични позиции на сайта. В следващите две фигури са показани съответно страницата за редакция на статии след добавяне на добавката и страницата „Контакти” на сайта след добавяне на карта.



Фиг. 3.15.. - Страница за редакция на статии след добавяне на добавката „Button - NoNumber Modules Anywhere”



Фиг. 3.16 - Страница „Контакти” на сайта след добавяне на картата.

### 3.3.2. Създаване на „Галерия”

За създаване на галерията беше избран модулът joombig image gallery vertical bar от <http://extensions.joomla.org/extension/joombig-image-gallery-vertical-bar>. Този модул беше избран понеже показва избрана папка със снимки в слайдшоу. Единствения недостатък на модулът е, че безплатната версия не позволява премахването на линка с авторското право. Този недостатък беше

отчетен като незначителен, на фона на таксата която би се платила за платена версия. Модулът joombig image gallery vertical bar беше добавен към статията по същия начин и поради същите причини като модула Branch Locations Google Map, чрез бутона Module в страницата за редакция на страници създаден с помощта на добавката Button - NoNumber Modules Anywhere. Страницата „Галерия” след добавяне на модула joombig image gallery vertical bar е показана на Фигура 3.17.



Фиг. 3.17. – Страница Галерия след добавяне на модул joombig image gallery vertical bar

### **3.3.3. Създаване на регистрационна форма и задаване на права на потребителите от администрацията на детското заведение**

Първоначалната идея за създаване на регистрационна форма беше да бъде избрана такава, която позволява при регистрацията на нов потребител да се пише името на детето, чиито родител се регистрира и да се прави проверка в базата данни дали съществува дете с такова име. Ако съществува регистрацията да бъде успешна, ако няма такъв запис да не бъде допускана регистрацията. След

проверка в множество сайтове безплатна такава регистрационна форма не беше открита и беше оставена за ползване формата по подразбиране на Joomla.

За администрацията на детската градина беше предоставен достъп Редактор (Editor), който позволява да се редактират статиите, но нямат права да създават или изтриват такива. Това ниво на достъп беше предоставено по тяхно желание, понеже на администрацията се налага да правят постоянни промени по Седмичното меню и Таксите, но не притежават достатъчни компютърни познания за да им бъде разрешен достъп до създаване и изтриване на статиите.

### **3.4. Публикуване и поддръжка на сайта**

#### **3.4.1. Публикуване на сайта**

Стъпките за публикуване на сайт в интернет са:

- Регистриране на собствено домейн име.
- Наемане на сървър от хостинг компания.
- Качване на уеб страниците на сървъра.
- Регистрация на сайта в търсачки и директории

Домейнът е уникално име, което сочи към дадена интернет страница.

В интернет не могат да съществуват два напълно еднакви домейна (имена на уеб сайтове). Поради тази причина той трябва да бъде регистриран при компании, наречени регистратори.

За да са видими от интернет, страниците на сайта трябва да се намират на сървър - специализиран компютър, който съхранява интернет страници. Сървърът има задачата да изпрати нужната информация, когато към него е изпратена заявка за някоя страница.

Може да се превърне личен компютър в сървър с помощта на XAMPP. XAMPP е съвкупност от най-използваните и най-важни уеб приложения: Apache, MySQL, FTP, Mercury Mail Server и TomCat. Инсталационният интерфейс на XAMPP е бърз и лесно достъпен. Но разбира се това има своите недостатъци. Когато компютъра се използва като сървър, не винаги може да се

подсигури непрекъснатата му свързаност към ток и интернет. При стартиране на странични приложения може да се повлияе върху работата на сървъра.

В днешно време редица фирми, наричани хостинг компании, предлагат сървъри под наем, на не много високи цени. Тези хостинг компании разполагат с мощни сървър компютри, много бърза и надеждна интернет връзка и компютърни програми, които улесняват публикуването и поддръжката на сайтове.

За настоящата дипломна работа бяха разгледани няколко хостинг компании. Избран беше персонален хостинг план на компанията host.bg. Този план беше избран на първо място заради предложената отстъпка и на второ отговаря изцяло за нуждите на сайта. На Фигура 3.18 е показан избрания хостинг план.

Персонален Премиум Бизнес Сравнение на плановете

**Хостинг**

### Персонален план от host.bg

Най-изгодния план за блог или малък фирмен сайт. Въпреки ниската цена разполагате с Контролен панел на български и възможност да инсталирате над 300 приложения само с 1 клик.

- 20 GB дисково пространство
- 20 допълнителни сайтове (Add-on домейни)
- 150 GB месечен трафик
- 300 безплатни инструмента
- MySQL5 на SSD
- Неограничени поддомейни
- Неограничен брой E-mail кутии
- Ежедневен backup
- Безплатен SSH достъп
- 24 / 7 / 365 поддръжка
- Опция: **SPAM EXPERTS**

**2.99\***  
лв./м  
Подновяване  
6.25\* лв./м

**Поръчай сега**  
(55% отстъпка)

- Сравнение на плановете
- Виж по-мощен план
- Свържи се с нас

Този сайт използва бисквитки (cookies) за предоставяне на услугите в него. За повече информация запознайте се с [условията за ползване](#).

Фиг.3.18. – Хостинг план на HOST BG

Хостинг компанията поддържа и регистрация на домейн име и двете стъпки (регистрация на собствено домейн име и наемане на сървър) бяха направени едновременно.

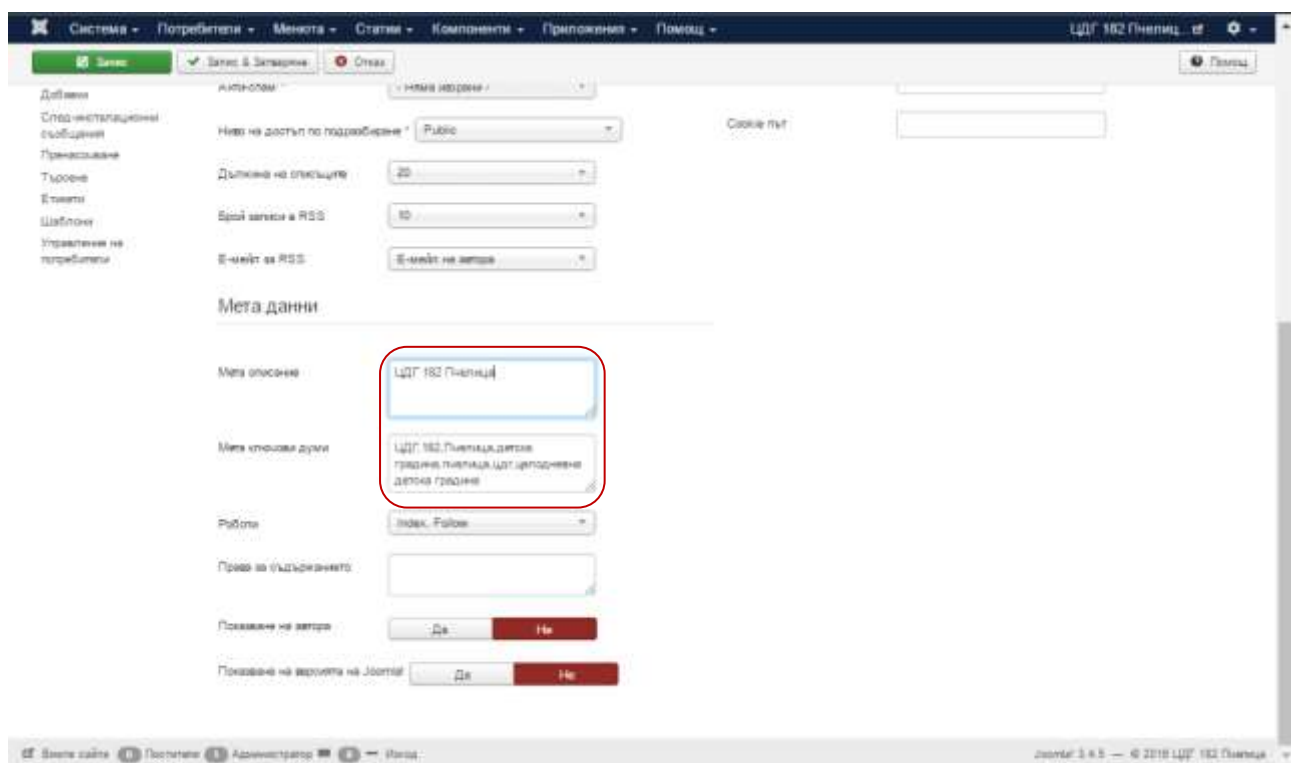
След получаване на данните за достъп от хостинг компанията сайта беше качен на сървъра и вече имаше напълно функциониращ сайт в Интернет пространството.



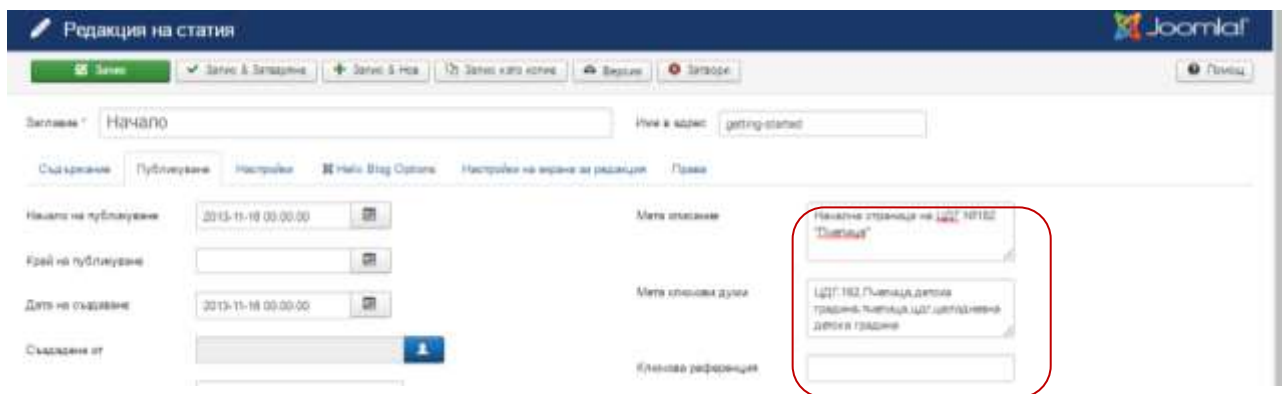
Следващата стъпка (регистрация на сайта в интернет търсачки) не е задължителна за съществуването на интернет сайт. Но ако се иска сайта да е достъпен при търсене в интернет търсачки като Google, Yahoo! и т.н., ще трябва да се направи регистрация на сайта към тези търсачки.

Преди да се направи това обаче, трябва да се оптимизира сайта за търсачки с така наречената SEO оптимизация. Това е набор от похвати, чиято цел е добро класиране на сайт в страницата с резултати при търсене по ключови думи в интернет търсачка. Ключовите думи (meta keywords), които са указани в сайта, се използват от интернет търсачките при определяне значимостта на страницата за търсене по ключови думи. Колкото по-точно е съответствието между ключовите думи на страницата и ключовите думи, по които е зададено търсене един потребител, толкова по-напред се класира страницата в списъка с резултати.

Задаване на ключови думи трябва да се направи както в глобалните настройки на сайта, така и на всяка статия. Задаването на ключови думи в глобалните настройки е показано на Фигура 3.19 , а задаване на ключови думи за страница на Фигура 3.20.



Фиг. 3.19 - Задаване на ключови думи в глобалните настройки



Фиг. 3.20. - Задаване на ключови думи за страница „Начало”

Когато сайтът е оптимизиран, трябва да се съобщи на търсачките за неговото съществуване – може да се подадете заявка индивидуално към всяка от интернет търсачките. В следващата таблица е предоставен списък с адреси за регистрация в някои от най-големите търсачки.

Таблица 3.8. - Списък с адреси за регистрация в някои търсачки

Име на търсачка	URL адрес за регистрация
Google	<a href="https://www.google.com/webmasters/tools/submit-url?pli=1">https://www.google.com/webmasters/tools/submit-url?pli=1</a>
Yahoo	<a href="http://siteexplorer.search.yahoo.com/au/free/submit">http://siteexplorer.search.yahoo.com/au/free/submit</a>
Bing	<a href="http://www.bing.com/toolbox/submit-site-url">http://www.bing.com/toolbox/submit-site-url</a>

### 3.4.2. Поддържане и актуализиране

Поддръжката и актуализирането на информацията във сайта са много важни за успешното присъствие в Интернет. Необходимо е сайтът постоянно да съдържа актуална информация за дейността на детската градина.

Поддръжката на сайта включва:

- актуализиране на текста в съществуващите страници;
- подмяна и добавяне на нови снимки;
- добавяне на нови уеб страници;
- редактиране на съществуващи страници, текст и графика;
- премахване на стари страници.

Поддръжката и актуализирането на информацията във сайта се извършва от авторката на настоящата дипломна работа.

Също така беше проведено обучение на персонала на детската градина за начина по който се редактират статии и се добавят снимки.

## РЕЗУЛТАТИ И ИЗВОДИ ОТ ТРЕТА ГЛАВА

В трета глава на настоящата дипломна работа беше създаден уеб сайт на детска градина №182 „Пчелица”. За целта бяха разгледани съществуващи сайтове на други детски заведения и им беше направен анализ за да може да се отделят задължителните изисквания при създаване на сайт за детска градина. Бяха изведени изискванията поставени от самата детска градина за това какво искат да има в сайта и как да изглежда. Също така бяха разгледани подробно стъпките по които се минава за създаването на какъвто и да е уеб сайт от избор на темплейт, избор на навигационно меню, изграждането на страници, добавянето на модули и добавки за нуждите на сайта до раздаването на права. Беше разгледан начина за публикуване на сайта в интернет пространството, както и какво се изисква за поддръжката на сайта след като е качен. Накрая вече имаше един напълно функциониращ уеб сайт.

## ЗАКЛЮЧЕНИЕ

Представената дипломна работа отразява цялостния жизнен цикъл на изграждането на един уеб сайт. Като най – основни фази могат да бъдат изведени следните:

- Избор на технология за неговото създаване
- Запознаване с подобни и/или конкурентни фирми/организации
- Определяне на изискванията
- Проектиране на дизайна, съдържанието и функционалностите
- Създаване, публикуване и поддръжка на сайта

Като обобщение може да се каже, че всяка зад всяка основна фаза от жизнения цикъл на уеб сайта стоят десетки други. Всяка от тях е от изключително значение за оформянето на стойностен краен продукт. Всяка една страница в създадения уеб сайт трябва да бъде добре обмислена, правилно структурирана и да носи търсената информация на потребителите.

Въз основа на свършеното в трите логически части може да се твърди, че целта на дипломната работа е изпълнена. За достъп до крайния продукт от разработката (сайт на ЦДГ №182 „Пчелица”) е необходимо да се използва адреса: <http://www.cdg182pchelica.com/>

## ИЗТОЧНИЦИ

### Литературни източници:

- [1] Браян Керниган, Денис М. Ричи, Програмният език С, 2008г (19.05.2016г.)
- [2] Катя Рашева – Йорданова, Контент мениджмънт системи Част 1: Теоретични аспекти, издателство „За буквите – о писменехъ“ 2014г (15 - 30.05.2016г.)
- [3] Светлин Наков и колектив, Въведение в програмирането с Java, 2008 г (18.05.2016г.)
- [4] Светлин Наков и колектив, Програмиране за .net framework, издателство "фабер", 2005 г. (19.05.2016г.)

### Онлайн ресурси:

- [5] Въведение в XML [Онлайн ресурс] (17.05.2016г.) - <http://uroci.net/urok/2044/%D0%92%D1%8A%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5-%D0%B2-XML.html>
- [6] Изработка на сайт [Онлайн ресурс] (16.05.2016г.) - <https://www.nit.bg/izrabotka-sajt>
- [7] Интернет технологии [Онлайн ресурс] (20.05.2016г.) - <http://tuj.asenevtsi.com/IntT/IntT054.htm>
- [8] Как се прави сайт №1 [Онлайн ресурс] (21.05.2016г.) - <http://www.parvaisit.com/css-introduction.html>
- [9] Какво може да прави PHP? [Онлайн ресурс] (19.05.2016г.) - <http://php.net/manual/bg/intro-whatcando.php>
- [10] Основи на визуалното програмиране. VisualBasic [Онлайн ресурс] (18.05.2016г.) - [http://bankova.ucoz.com/Programirane/osnovi\\_na\\_vizualnoto\\_programirane-2.pdf](http://bankova.ucoz.com/Programirane/osnovi_na_vizualnoto_programirane-2.pdf)
- [11] Програмиране в среда Интернет [Онлайн ресурс] (20.05.2016г.) - <http://www.mgu.bg/drugi/ebooks/belchevski/perl.html>
- [12] Програмиране за iOS [Онлайн ресурс] (20.05.2016г.) - [https://www.fmi.uni-sofia.bg/education/bachelors/izbiraemi\\_11\\_12\\_zimen/Programirane%20za%20iOS.pdf](https://www.fmi.uni-sofia.bg/education/bachelors/izbiraemi_11_12_zimen/Programirane%20za%20iOS.pdf)
- [13] Увод в Програмирането [Онлайн ресурс] (16.05.2016г.) - <http://informatics.smg.bg/uploads/2015-10-09/magdalena.todorova-cplus.programirane.pdf>
- [14] Уеб дизайн и изработка на динамични сайтове [Онлайн ресурс] (17.05.2016г.) - <https://violleta.com/internet-termini/ueb-dizain>

- [15] Факултет по математика и информатика - Въведение в езика ActionScript MX за Flash MX [Онлайн ресурс] (20.05.2016г.) - <http://fmi-plovdiv.org/index.jsp?id=1137&ln=1>
- [16] Що е то Java? [Онлайн ресурс] (18.05.2016г.) - <http://uroci.net/forum/index.php?showtopic=11777>
- [17] ActionScript - общи сведения [Онлайн ресурс] (20.05.2016г.) - <http://bestwebdesign.org.uk/%D1%83%D1%80%D0%BE%D1%86%D0%B8/actionscript/148-actionscript-%D0%BE%D0%B1%D1%89%D0%B8-%D1%81%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D1%8F.html>
- [18] C++ [Онлайн ресурс] (19.05.2016г.) - <http://progressbg.net/c-kurs/>
- [19] Perl web build bBulgaria [Онлайн ресурс] (21.05.2016г.) - <http://web-buildbg.com/perl/>
- [20] PHP - The Right Way [Онлайн ресурс] (19.05.2016г.) - <http://bg.phptherrightway.com/>
- [21] Python [Онлайн ресурс] (20.05.2016г.) - <http://vanyog.com/solutions/python.php>
- [22] Ruby - най-добрият приятел на програмиста [Онлайн ресурс] (20.05.2016г.) - <https://www.ruby-lang.org/bg/about/>
- [23] Web сървър [Онлайн ресурс] (17-20.05.2016г.) - <http://www-it.fmi.uni-sofia.bg/courses/WDB/ws.htm>
- [24] WordPress - ръководство, информация, уроци, полезни съвети [Онлайн ресурс] (21.05.2016г. ) - <https://rykovodstvo.wordpress.com/2013/09/05/%D1%81%D1%8A%D1%89%D0%BD%D0%BE%D1%81%D1%82-%D1%85%D0%B0%D1%80%D0%B0%D0%BA%D1%82%D0%B5%D1%80%D0%B8%D1%81%D1%82%D0%B8%D0%BA%D0%B8-%D0%B8-%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BD%D0%B0/>
- [25] 10 езика за програмиране, които трябва да научите [Онлайн ресурс] (18-22.05.2016г.) - <http://ureport.bg/12833/2014/05/21/obshtestvo/10-ezika-za-programirane-koito-tryabva-da-nauchite>