

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Informaatikainstituut

Tarkvaratehnika õppetool

**Sisuhaldussüsteemi valimine veebilehe
loomiseks lähtudes tellija soovidele ning
kasutades lihtsaid ja efektiivseid
lahendusi.**

Bakalaureusetöö

Üliõpilane: Ilya Kapitan

Üliõpilaskood: 112674IAPB

Juhendaja: Jekaterina Ivask

Tallinn
2015

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

(kuupäev)

(allkiri)

Annotatsioon

Käesolev bakalaureuse lõputöö on pühendatud lihtsale ja mugavale veebisaidi kasutamisele, mis on suunatud kliendi soovide täitmisele.

Eesmärgi saavutamiseks selgitati välja vajalik informatsioon tellitava veebilehe kohta. Saadud andmete põhjal pandi kokku veebilehe funktsionaalsed ja mitte-funktsionaalsed nõuded.

Interneti keskkonnas viidi läbi olemasolevate sisuhaldussüsteemide analüüs. Valiti kõige sobivam süsteem - WordPress, sest veebliehe arendamisel on see arusaadav ja lihtne platvorm. WordPressi mugavus väljendub ka hilisemal lehe haldamisel, kuna ta võimaldab kliendil teha vajalikke toiminguid iseseisvalt.

Läbi sai töötatud ka süsteemi funktsionaalsus vastavalt tellimusele ja rakendati hea kasutajaliides lehe lihtsaks ja mugavaks kasutamiseks. Enne teema ja disaini loomist sai valitud raamistik, millele ehitati kasutajaliides. Parima tulemuse saavutamiseks uuriti põhjalikult veebilehtede loomise põhimõtteid.

Töö täitmise käigus loodi mitmekeelne veebileht, koos võimalusega tellida 3D printimist ja tellimuse täitmise jälgimise teenust. Lisaks viidi läbi veebilehe optimeerimine optimaalse tööressursi saavutamiseks.

Lõputöö on kirjutatud vene keeles ning sisaldab teksti 46 leheküljel, 6 peatükki ja 12 joonist, 7 tabelit.

Abstract

This bachelor's thesis is devoted to the creation of a simple and easy-to-use web site, aimed at the realization of the customer's wishes.

To achieve the current goal, information regarding wished website was primarily received.

Based on the provided data identification of functional and non-functional requirements for the website was formulated. Analysis of the existing content management systems on the Internet was made. Most suitable system – WordPress, was chosen. This system is easy to understand and develop, as well as easy to manage, allowing the customer to make the needed operations without much trouble.

Also, the functionality of the system has been modified to comply with the order and a nice and simple user interface was implemented. Before creating a theme and a design, a framework had been selected, based off of which the interface was created. For receiving best results, principles of creating a website were studied.

In the course of performing this thesis, a multilingual website with the possibility of purchasing 3D printing and tracking your order services, was created. Furthermore, in order to achieve optimal resource performance, web site optimization was performed.

The thesis is in russian and contains 46 pages of text, 6 chapters, 12 figures, 7 tables.

Аннотация

Дипломная работа посвящена созданию простого и удобного в использовании веб-сайта, направленного на реализацию желаний заказчика.

Для достижения данной цели прежде всего была получена информация о желаемом сайте. По этим данным были сформулированы функциональные и нефункциональные требования. Был проведен анализ существующих систем управления контентом в интернете. Была выбрана наиболее подходящая система – WordPress, так как она проста в понимании и легкая для разработки, а также удобна в управлении, что позволит заказчику производить нужные ему операции без особого труда.

Также был доработан функционал системы для соответствия заказу и реализован приятный и простой пользовательский интерфейс. Перед созданием темы и дизайна был выбран фреймворк, на основе которого делался интерфейс. Для достижения оптимальных результатов были изучены принципы построения страниц.

В ходе выполнения работы был создан многоязычный веб-сайт, с возможностью заказа услуг 3D печати и отслеживания своего заказа. Кроме того, была проведена оптимизация сайта, чтобы достичь оптимальной работы ресурса.

Дипломная работа написана на русском языке, содержит 46 страниц текста, 6 глав, 12 рисунков, 7 таблиц.

Список сокращений и терминов

Веб-страница	<i>Web page/website</i> Документ, который просматривается посредством веб-браузера. Содержание пригодно к обработке и изменениям[1].
HTML	<i>HyperText Markup Language</i> Набор символов разметки, предназначенных для отображения веб-страницы в браузере[2].
CSS	<i>Cascading Style Sheets</i> Язык описания внешнего вида документа/веб-страницы, написанной с использованием набора символов разметки[3].
Скрипт	<i>JavaScript</i> Язык программирования для придания более красочного вида и интерактивности веб-страницам[4].
Фреймворк веб-страницы	<i>Front-end Framework</i> Каркас или основа, предоставляющая общую функциональность для создания веб-страницы[5].
SEO	<i>Search Engine Optimization</i> Меры для поднятия позиций сайта в результатах выдачи поисковых систем по разным запросам с целью продвижения сайта[6].
Домен	<i>Domain</i> Символьное имя, служащее для идентификации областей в сети Интернет[7].
Поддомен	<i>Subdomain</i> Домен, являющийся частью другого домена более высокого уровня[8].

Плагин	<i>Plugin</i> Независимый модуль, динамически подключаемый к основной системе для расширения функционала[9].
XML	<i>eXtensible Markup Language</i> Язык разметки гипертекста для документов, содержащих структурированную информацию[10].
Пользовательский интерфейс	<i>User Interface</i> Система правил и средств, регламентирующая и обеспечивающая взаимодействие программы с пользователем[11].
3D	<i>Three-dimensional Space</i> Трехмерное пространство – модель материального мира, в котором мы находимся[12].
STL	<i>STereoLithography</i> Формат файла, используемый для хранения трехмерных моделей для использования в печати[13].
CMS	<i>Content Management System</i> Система управления содержимым, используется для организации процесса создания, редактирования и управления содержимым[14].
Тэг	<i>Tag</i> Элемент разметки гипертекста[15].

Список иллюстраций

Рисунок 1. Доля рынка у разных движков[16].	16
Рисунок 2. Список активированных языков для сайта.	26
Рисунок 3. Добавление страницы, с возможностью добавления перевода.	26
Рисунок 4. Адаптивный дизайн, пример отображения информации[24].	28
Рисунок 5. Количество точек на дюйм стандартного дисплея и retina[26].	29
Рисунок 6. Вид изображения на обычном экране и retina экране[27].	29
Рисунок 7. Оптимизация изображений под Retina экраны[28].	30
Рисунок 8. Оптимизация основных изображений для сайта.	32
Рисунок 9. Главная страница на разных устройствах.	35
Рисунок 10. Персональная страница пользователя.	36
Рисунок 11. Изменение персональных данных.	37
Рисунок 12. Возможность добавлять STL модель при оформлении заказа.	38

Таблица фрагментов кода

Фрагмент кода 1. Изменения тэга input для принятия STL файлов.	25
Фрагмент кода 2. Команда в .htaccess файле для активации Gzip компрессии.....	33
Фрагмент кода 3. Команда в .htaccess файле для ликвидации механизма ETags.....	33
Фрагмент кода 4. Команда в .htaccess файле для отключения хотлинкинга.	33
Фрагмент кода 5. Команда в .htaccess файле для контроля времени хранения кэша.	34
Фрагмент кода 6. Конечный файл .htaccess, написанный автором.	45
Фрагмент кода 7. Файл robots.txt, написанный автором.....	46

Содержание

1. Введение	12
1.1 Описание проблемы	12
1.2 Постановка задач	13
1.3 Методика	13
1.4 Обзор работы	14
2. Требования к веб-сайту.....	15
2.1.1 Функциональные требования.....	15
2.1.2 Нефункциональные требования.....	15
3. Выбор движка	16
3.1 Drupal	17
3.2 Joomla.....	18
3.3 Wordpress.....	20
3.4 Конечный выбор.....	22
4. Описание реализации	24
4.1 Магазин и пользовательские возможности.....	24
4.2 STL файл при оформлении заказа.....	25
4.3 Многоязычность	25
4.4 Создание темы	26
4.4.1 Адаптивный веб-дизайн.....	27
4.4.2 Progressive Enhancement или Graceful Degradation.....	28
4.4.3 Retina.....	29
5. Полученная реализация	31
5.1 Производительность.....	31
5.1.1 Эффективный плагин кэширования	31
5.1.2 Оптимизация изображений.....	31
5.1.3 .htaccess.....	32
5.2 Пользовательский интерфейс.....	35
5.3 SEO.....	38
6. Заключение.....	39
Kokkuvõte	40

Summary.....	41
Использованная литература	42
Дополнение	45

1. Введение

Данная тема очень актуальна, поскольку в наше время люди проводят все свое свободное время в интернете. Они заходят на интернет страницы, чтобы найти интересующую их информацию или услуги. Нынешнее поколение настолько лениво, что им проще найти товар и заказать что либо через интернет. Поэтому сейчас очень важно предоставлять им такую возможность на высоком уровне.

Для этого существуют всевозможные решения и технологии. Какие-то из них старые, какие-то новые, некоторые работают быстрее других и выявить более эффективные не так просто.

Автору данной работы, будучи веб разработчиком, приходилось ни раз сталкиваться с проблемами выбора оптимальных решений при создании веб-страниц. При получении нового заказа на создание сайта, автор решил изучить современные технологии и применить их на практике, чтобы добиться наиболее лучших результатов.

1.1 Описание проблемы

В настоящее время интернет индустрия находится на пике развития. Из-за большой конкуренции становится труднее удерживать посетителей на собственных ресурсах. Владельцы сайтов прибегают к необычным и интересным решениям, чтобы хоть как-то выделиться и привлечь к себе новых пользователей.

Современные страницы включают в себя много дополнений: таблицы, изображения, видео, формы оформления заказа, панели управления и т.д. Все это требует больше вспомогательных решений, таких как плагины и скрипты, купленных в интернете или сделанных своими руками.

Нередко происходят случаи, когда веб-страница работает в одном браузере, а в другом нет. Страница может отличаться внешним видом в разных браузерах. Кроме того, с популярностью умных устройств, таких как, телефоны, планшеты, телевизоры и часы, появляется больше трудностей, так как сайт перестает работать на тех или иных устройствах. Скорость загрузки страниц всегда остается актуальной проблемой.

Сложности добавляет еще и тот факт, что тематику для создания быстрой и технологичной веб-страницы автору дал заказчик и сайт предназначался не для персонального пользования, а для коммерческих целей, а именно для компании занимающийся 3D печатью.

К сожалению, 3D печать сравнительно новая индустрия и готовых веб-страниц сейчас почти нет, а существующие способны решить только часть проблем. Так как такие проекты уникальны – единого универсального решения не существует.

1.2 Постановка задач

Основной целью данной работы является создание сайта, отвечающего нуждам клиента. Клиент заказал многоязычный сайт услуг по 3D печати. На сайте должна быть форма заказа, с возможностью добавления файла 3D модели, страница с контактными данными и пользовательская страница, с возможностью следить за своими покупками и изменять персональные данные. Данный веб-сайт должен быть быстр, прост и привлекателен, чтобы у посетителя сайта возникало желание изучить продукт и возможно воспользоваться услугами сайта.

Сайт нужно сделать настолько простым, чтобы после передачи проекта заказчику, владелец сайта мог бы сам справиться с внесением простых изменений на страницах и добавлять/изменять изображения и текст.

1.3 Методика

Для создания сайта с наиболее эффективной работой необходимо проанализировать существующие интернет платформы. Платформа значительно упростит реализацию сайта, так как она включает в себя множество функциональных возможностей, что значительно экономит время.

Для разработки сайта нужно составить необходимые требования и на основе полученной информации выбрать платформу. Необходимо также учесть, что платформа может не решить всю задачу, скорее всего придется функционал дорабатывать.

В качестве языков для скриптов, внешнего вида и других решений были выбраны наиболее стандартные языки, используемые при интернет-разработке: PHP, HTML, CSS, JavaScript.

При разработке сайта необходимо учесть, что он может посещаться с разных устройств и вид должен не сильно отличаться, нужно сохранять стилистику сайта на всех видах устройств.

1.4 Обзор работы

Данная работа делится на 4 основные части.

В первой части описаны функциональные и нефункциональные требования к веб-сайту.

Во второй части описывается выбор движка. Приводится перечень наиболее популярных систем, описание их ключевых особенностей и возможностей, а также причины выбора определенного движка для реализации проекта.

В третьей части описывается реализация проекта. Объясняется как производились доработки, описывается как и на каких принципах создавалась визуальная оболочка.

В четвертой части описываются полученные результаты. Говорится о том, как была произведена оптимизация сайта, как был создан пользовательский интерфейс и как проводилась поисковая оптимизация. Также в ней виден конечный продукт.

2. Требования к веб-сайту

На основе проанализированной информации данной заказчиком автор построил список требований к веб-сайту, которые он должен реализовывать.

2.1.1 Функциональные требования

Веб-страница должна:

- Иметь возможность оформление покупки
- Иметь возможность прикладывать файл 3D модели при заказе услуг
- Иметь возможность клиенту следить за статусом покупки
- Иметь возможность клиенту связаться через контактную форму
- Иметь потенциальную возможность завести блог

2.1.2 Нефункциональные требования

Веб-страница должна:

- Быть узконаправленной и решать чётко поставленные задачи, не предоставляя большого количества лишней функциональности
- Быть мультиязычной (работать на эстонском, русском и английском языках)
- Иметь дружелюбный для пользователя интерфейс (интуитивный и простой в использовании)
- Иметь возможность для расширения функциональности в будущем (например, блог)
- Корректно работать в современных браузерах и быть надёжной.

3. Выбор движка

Для определения основных проблем и задач для создания сайта был проведен анализ существующих технологий. В ходе работы автор сравнивает выявленные технологии и выбирает наиболее эффективные решения. Основной анализ проводился для выявления подходящей CMS системы для решения поставленной задачи.

Движок или CMS – это сердце сайта, вокруг которого построены все механизмы. CMS, в переводе с английского система управления содержимым, должна быть простой в использовании как для программиста, так и для пользователя, помимо этого система должна быть гибкая, с возможностью дооснащения, так как в большинстве случаев сама система является каркасом с базовым набором «жизненно» необходимых возможностей. Для выбора движка было проанализировано множество разных систем и охваченные ими доли рынка.

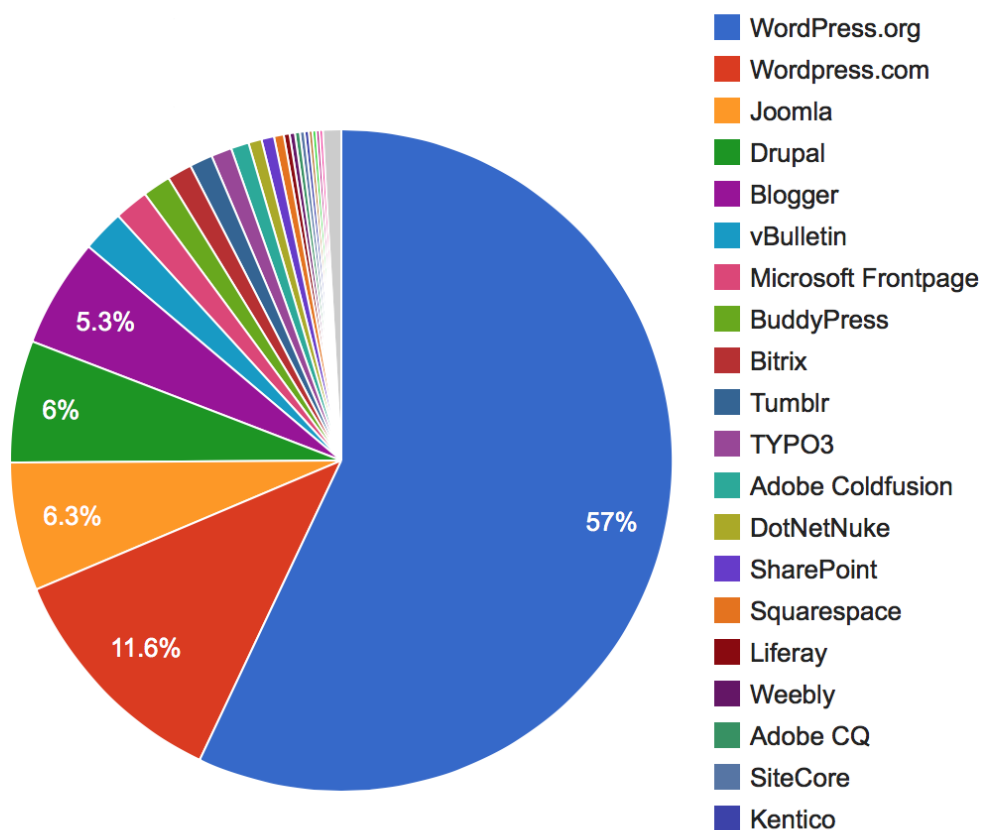


Рисунок 1. Доля рынка у разных движков[16].

Как видно из рисунка наиболее популярные платформы это WordPress, Joomla и Drupal.

Три эти системы существуют уже давно и прошли долгий путь в этапе развития. Их объединяет и тот факт, что все они открыты для бесплатного доступа, как open-source проект и развиваются благодаря множеству пользователей.

3.1 Drupal

Drupal была основана в 1999 году и благодаря огромному интересу среди пользователей проект был открыт как open-source в течении того года. Система развивается по сей день, на данный момент Drupal имеет 7ю версию. Drupal считается самой серьезной CMS из трех и является более сложной.

Ключевые особенности и возможности[17]:

Безопасность

Drupal имеет самую высокую защиту среди своих конкурентов на данный момент. Команда по безопасности быстро выявляет проблемы и устраняет их, если таковые имеются. Сайт белого дома работает на Drupal, что подтверждает сложность взлома данной системы.

Персональные типы данных и видов

Drupal позволяет создавать любые тип данных, даже такие, которых в системе изначально не существует. Хотите опубликовывать рецепты, создавать приватные страницы пользователей, добавлять картины и следить за передвижением экспонатов, все это можно сделать в Drupal.

Пользователи и личные права

В Drupal можно создавать разные группы пользователей и задавать разным группам определенные права, которые нужны для работы вашего сайта.

Гибкость

У сайтов Drupal нет границ, можно создать что угодно, будь то обычный сайт галерея или сайт серьезной компании. Вы можете создать простой сайт и потом дополнить его

функционалом и возможностями, Drupal расширяется и делает это с такой легкостью, что пользователь даже не понимает, как это происходит.

Масштабируемость

Сайты под управлением Drupal легко справляются с масштабностью. Система способна справиться с увеличением работы и повышением трафика. Даже миллион пользователей в день не подвергнут работу сайта под управлением Drupal.

Категории

Drupal обладает многофункциональной системой категорий, которая позволяет делать организацию содержимого и ставить тэги на содержимое. Каждая группа категорий может иметь специальные ограничения по персональным типам данных. Тэги можно так же настраивать по этим же принципам.

Интернет коммерции

У Drupal имеется коммерческое расширение Drupal Commerce. Это расширение позволяет создать любое решение связанное с коммерцией, можно сделать интернет магазин, сервис с подпиской, систему пожертвований для поддержки некоммерческих организаций.

Drupal сообщество

Drupal опирается как open-source решение. Это означает, что любое дополнение для системы можно достать бесплатно или почти за бесплатно. На <http://drupal.org> можно найти все о чем вы пожелаете, пользователи этой системы настоящие патриоты и экспериментаторы, даже такие решения создаются, которые еще не стали популярными, но в скором времени станут.

3.2 Joomla

Joomla начала зарождаться в 2000 году и только в 2002 году появилась первая бесплатная версия. Joomla является серединой между Drupal и Wordpress. Она более простая, чем Drupal, но и сложнее Wordpress'a. Joomla обладает подобной структурой, что и WordPress, но позволяет решить задачи с которыми труднее справиться.

Ключевые особенности и возможности[18]:

Простой интерфейс

Пользовательский интерфейс Joomla очень легкий и простой в понимании. Панель администрации понятна, без нужды писать дополнительные решения. Любой пользователь может войти в панель администрации, изучить и воспользоваться всем функционалом, предоставляемым Joomla.

Безопасность

Система Joomla обладает хорошей безопасностью. Она устроена так, что пользователь не сможет зайти в никакой каталог сайта используя прямую ссылку в адресной панели. Это делает все ваши файлы защищенными, без прямого доступа к ним через адресную строку.

Пользователи, типы, группы

Joomla предоставляет много типов пользователей на выбор, например зарегистрированный пользователь, автор и т.д. У Joomla есть так же множество групп пользователей, например: Публицист, Автор, Копирайтер, Зарегистрированный. Функционал, права групп и типов можно изменять.

Гибкость

При использовании системы Joomla вы можете сделать множество разновидностей сайтов: информационные сайты, познавательные порталы, государственные страницы, бизнес сайты, сайты развлечений, новостной портал или интернет магазин.

Расширения

Функционал Joomla можно дополнить используя расширения, они с легкостью устанавливаются без надобности что-то менять в системе. На сайте компании есть огромная библиотека расширений и большинство из них бесплатны.

Адаптивный дизайн

С выходом третьей версии Joomla был введен адаптивный дизайн. Это позволяет пользователям с легкостью управлять сайтом с телефоном и планшетов, без надобности использовать компьютер.

Настройка и установка

Joomla очень прост в установке и настройке. Чтобы установить сайт на сервер и настроить его нужно потратить 10-15 минут. Для обычных пользователей, которые не разбираются сильно в системе или не знают как настроить ее, есть возможность установки в одно нажатие и система сделает все за вас.

3.3 Wordpress

WordPress появилась в 2003 году. Изначально WordPress была сделана как платформа для блогеров. С тех пор она очень преобразовалась благодаря пользователям. Помимо продвижения системы WordPress открыли свой интернет сервис, где можно создать свой сайт, без необходимости арендовать место для размещения сайта (хостинг). Система бесплатная, код открыт open-source, это позволяет полностью изменять WordPress, расширять функционал и дополнять систему недостающими элементами.

Ключевые особенности и возможности[19]:

Простота использования

WordPress очень легок в использовании и имеет простой и понятный интерфейс. Добавление страниц, постов, картинок и т.д. занимает считанные секунды, а время, нужное на дальнейшее форматирование значительно сокращено, благодаря новым технологиям.

Гибкость

WordPress, благодаря своему функционалу, позволяет создать любой сайт: личный блог, фотоблог, бизнес сайт, портфолио, новостной портал, магазин или даже свое приложение.

Возможность работать с любого компьютера

WordPress – это автономная система, не нуждающиеся в дополнительных программах для редактирования HTML файлов, таких как Coda, Notepad++, Adobe Dreamweaver и т.д. WordPress изначально был создан на основе браузера, что позволяет пользователю зайти в интернет с любого компьютера и управлять сайтом. Все необходимые действия, добавление графики, страниц, постов, форматирование текста, добавление мета тэгов и т.д. Все это можно сделать через систему WordPress без дополнительных FTP, HTML программ.

Поисковые движки любят сайты, сделанные на WordPress.

У WordPress очень чистый и простой код, что упрощает чтение сайта для поисковых движков и позволяет легче индексировать сайт. Даже глава спам отдела в Google говорит, что WordPress это хороший выбор[20].

Кроме того, каждая страница, пост и картинка могут иметь свои личные описания, заголовки, альт текста (в случае графики), мета ключевые слова и т.д, что оптимизирует контент для определенных ключевых слов, позволяя для более точной оптимизации для поисковых движков. Можно так же использовать тэге для более лучшей поисковой работы.

Полный контроль над сайтом

Не надо ждать дизайнера, чтобы сделать простые изменения на сайте. WordPress позволяет управлять/изменять практически любую часть сайта, благодаря чему простые изменения могут быть сделаны без чьей либо помощи.

Дизайн сайта полностью «настраиваемый»

WordPress выступает в роли движка для сайта. Внешний вид сайта можно полностью создать и настроить, чтобы сайт был уникален и выражал черты бренда. Система WordPress основана на темах, что позволяет пользователям скачать или купить имеющиеся темы, в случае отсутствия своей темы.

Встроенный функционал блога

WordPress изначально создавалась как платформа для блоггеров, поэтому полный функционал для блога уже встроен изначально и возможен к активации, при желании.

Отобразить недавние публикации, включить комментарии, возможность подписки и RSS, все это очень просто настроить и включить, а в свою очередь блог поможет сделать сайт более интерактивным и увлекательным.

Возможность расширения функционала сайта при помощи плагинов

Плагины – это так называемые дополнения, которые расширяют функционал WordPress сайта и добавляют ему определенный функционал. Добавить календарь, видео галерею, twitter/facebook функционал и многое другое, всего этого можно добиться благодаря плагином, большинство из которых являются бесплатными, а те, что платные имеют разумную цену. GPL лицензия[21] позволяет изменять WordPress так, как клиент этого хочет.

Сайт растет вместе с бизнесом

WordPress сайты легко масштабируемы. На сайте могут быть сотни тысяч страницы и публикаций, но даже это количество не повлияет на производительность системы.

Управление пользователями

Как администратор WordPress сайта, вы можете назначать пользователям разные уровни доступа и возможности для этих уровней. Так же можно создавать, удалять и изменять пользователей и их данные. При использовании правильных дополнений, можно добиться полного контроля над пользователями и их действий.

Многоязычность

Система WordPress многоязычна, на данный момент она переведена на более чем 70 языков и при использовании дополнительных плагинов можно сделать многоязычный сайт.

3.4 Конечный выбор

Была выбрана система WordPress, так как на момент написания работы у автора уже имелся большой опыт работы с системой. Помимо этого именно WordPress отвечал нуждам для решения поставленной задачи:

- Система проста в использовании, соответственно даже необученный пользователь сможет справиться с управлением.
- Система позволяет добиться многоязычности.
- Пользователю легко управлять содержимым без надобности вносить изменения в программную часть сайта или его внешний вид.
- WordPress очень гибкая система и при правильных дополнениях позволяет добиться любой поставленной цели.
- WordPress сама по себе является платформой для блогов, тем самым решает одну из поставленных задач о потенциальном развитии сайта.
- Так как темы являются полностью настраиваемы, автору не придется постоянно менять код внешнего вида сайта, его можно будет легко настроить из панели администрирования.

4. Описание реализации

WordPress – мощная система управления контентом с большим функционалом, но даже она, в базовой комплектации, не способна решить поставленную автором задачу.

Чтобы добиться поставленных целей необходимо было решить следующие проблемы:

- Функционал магазина
- Пользовательская страница, с возможностью просмотра истории заказов и изменениями данных
- Поддержка STL файлов и возможность приложить файл при оформлении заказа
- Многоязычность
- Тема

4.1 Магазин и пользовательские возможности

Как и говорилось ранее, у WordPress отсутствует функционал магазина, по умолчанию, но пользовательский функционал существует. Есть возможность зарегистрироваться и входить в свою учетную запись, выходить из нее, но нет конкретной панели управления, где можно было бы выполнять разные операции.

WooCommerce – один из самых популярных плагинов для электронной торговли в WordPress. WooCommerce включает в себя много возможностей, автор воспользовался следующим функционалом:

- Продажа услуг
- Статистика
- Выставление счета
- Управление инвентарем

WooCommerce так же реализует профиль пользователей и историю покупок.

4.2 STL файл при оформлении заказа

Функционал WooCommerce можно расширять дополнительными плагинами. Product Add-ons – это плагин позволяющий дополнить форму заказа для клиентов, можно добавить поля для ввода, параметры на выбор или приложение файла.

Плагин изначально создавался под общие форматы файлов, STL формат недоступен в этом дополнении. Для решения проблемы нужно было не только сделать возможность добавлять файлы такого формата, но и сделать их единственными возможными файлами для приложения, иначе клиенты будут присылать разные форматы файлов и это будет мешать работе сайта. Этому автор добился внося простые изменения в коде плагина, тэг input. У input есть такой параметр как accept, он отвечает за то, какие файлы можно добавлять в форму.

Фрагмент кода 1. Изменения тэга input для принятия STL файлов.

```
<input type="file" accept=".stl">
```

Так как у accept указан STL формат, то другие форматы выбрать невозможно. Этот параметр появился в версии HTML5, в более старых браузерах он не поддерживается. В данной работе автор преследовал только новые технологии, поэтому поддержка более старыми браузерами осталась нерешенной.

4.3 Многоязычность

Не смотря на то, что WordPress является многоязычным сам по себе, сайт на нескольких языках он не позволяет сделать. Есть разные варианты, чтобы добиться многоязычности. Автор решил эту проблему плагином WPML, специализирующийся на переводах, выполненных людьми. WPML (The WordPress Multilingual Plugin) – самый популярный и наиболее функциональный плагин в сфере переводов для WordPress на данный момент. При работе с этим плагином, все переводы происходят на одном сайте с возможностью переключать язык.

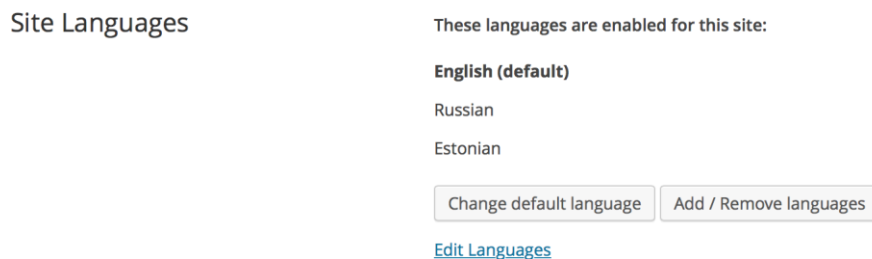


Рисунок 2. Список активированных языков для сайта.

На рисунке номер 2 видно часть настроек плагина, а именно выбранные автором языки для данной работы.

При написании статьи или добавлении страницы есть возможность написать ее и на другом языке, плагин понимает, что это одна и та же страница на разных языках и дает ее редактировать, как это видно на рисунке номер 3.

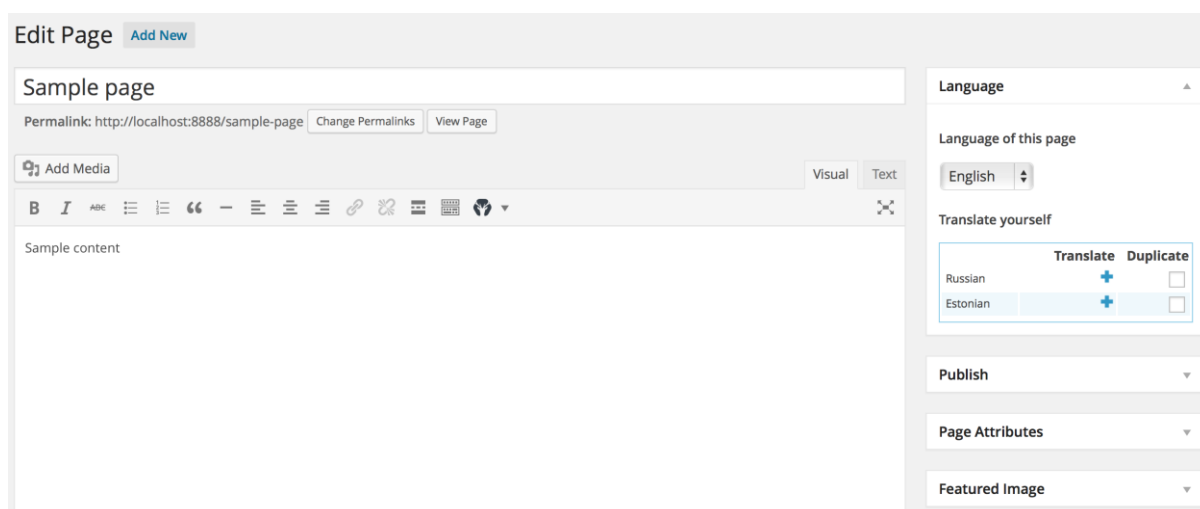


Рисунок 3. Добавление страницы, с возможностью добавления перевода.

4.4 Создание темы

WordPress использует систему тем, для изменения внешнего вида сайта. Такое решение сделано по нескольким причинам:

- Тема WordPress разделяет стили и файлы темы от системных файлов, тем самым позволяя обновлять систему WordPress и ее компоненты, не нарушая работы темы.

- Раздельность файлов обеспечивает безопасность содержимому сайта, так как созданные страницы и публикации не пропадают при смене дизайна.
- Темы позволяют производить быструю смену дизайна путем пары нажатий.
- Система тем позволяет менять внешний вид даже неопытным пользователям.

Пользователь может скачать одну из бесплатных тем, купить более качественную или же создать свою.

Создание темы для WordPress начинается путем верстки HTML шаблона, пишутся стили и делается каркас, а потом уже все это дополняется функционалом самой системы. Чтобы не писать все решения с нуля, автор использовал html/css фреймворк Foundation Zurb[22] для создания дизайна сайта, потому что он open-source и очень популярен сейчас.

Опираясь на фреймворк Foundation, автор следовал следующим аспектам:

- Адаптивный дизайн (responsive design)
- Progressive Enhancement или Graceful Degradation
- Retina

4.4.1 Адаптивный веб-дизайн

Впервые понятие адаптивный веб-дизайн появилось в 2010 году, написал об этом Итан Маркотт (Ethan Marcotte)[23].

Целью адаптивного веб-дизайна является универсальность и корректное отображение информации на различных устройствах. Страница подстраивается под размеры экрана и поэтому один сайт может работать на телефоне, на планшете, на компьютере и даже телевизоре, если там есть интернет.

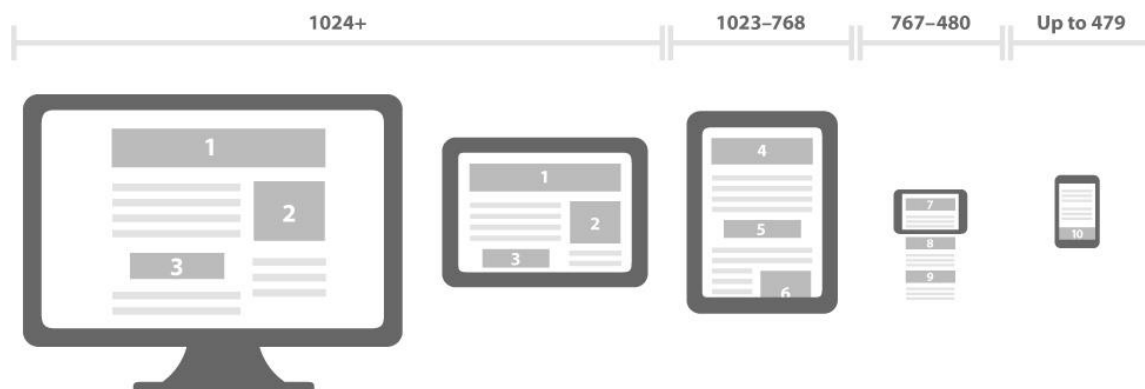


Рисунок 4. Адаптивный дизайн, пример отображения информации[24].

На рисунке номер 4 показано, как информация выводится в зависимости от размеров экрана. Вывод информации на устройствах зависит также от методики, используемой при разработке.

4.4.2 Progressive Enhancement или Graceful Degradation

Понятие «Progressive Enhancement» впервые появилось в 2003 году и сформулировал его Steven Champeon[25]. Этот принцип заключается в создании сначала простой семантической HTML-верстки для представления всей информации, а дальше уже в последствии по мере улучшения браузера, развивать и дополнять функционал, такой как CSS и JavaScript.

«Graceful Degradation» является традиционным методом разработки в наше время, противоположным «Progressive Enhancement». При разработки проекта по такому принципу ориентируются на самые современные браузеры, а поддержка старых браузеров и корректность работы уже дорабатывается потом.

Хоть Progressive Enhancement является и более новым методом, автор выбрал для реализации сайта метод «Graceful Degradation», проект проще продумать в полном масштабе с начала, чем начинать с нуля и развивать до конца, и так как реализуется маленький проект, то метод Progressive Enhancement тут не нужен, потому что клиенты из сферы 3D печати пользуются более современными устройствами.

4.4.3 Retina

Retina Display – это бренд, используемый компанией Apple для своих экранов, у которых более высокая плотность пикселей в сравнении с обычным экраном. Apple это компания гигант с невероятным количеством пользователей, поэтому с ее изобретениями нужно считаться и оптимизировать свои ресурсы под них.

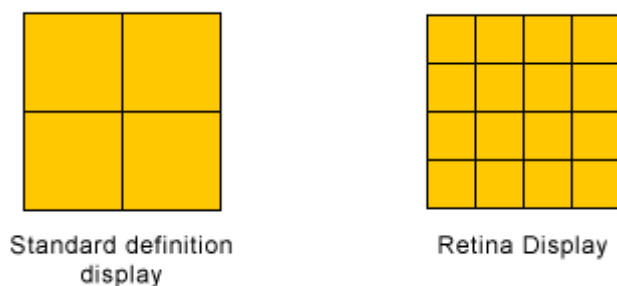


Рисунок 5. Количество точек на дюйм стандартного дисплея и retina[26].

Разрешение у таких дисплеев выше, чем у full HD, плотность пикселей у Retina экрана в 2 раза больше, чем у простых экранов, это показано на рисунке номер 5. Из-за этого если близко взглянуть на изображение на простом экране, как видно на рисунке номер 6, оно покажется нечетким, в сравнении с Retina экраном.



Рисунок 6. Вид изображения на обычном экране и retina экране[27].

Для реализации отображения retina изображений автор установил дополнительный скрипт на сайт под названием retinaJS. Он работает так, что при загрузке сайта проверяется какой дисплей у пользователя и если дисплей retina, то обычные изображения заменяются на изображения @2x, если они имеются. Изображение @2x это обычное изображение увеличенное в 2 раза. На рисунке номер 7 показано, как нужно оптимизировать обычное изображение под Retina экран.

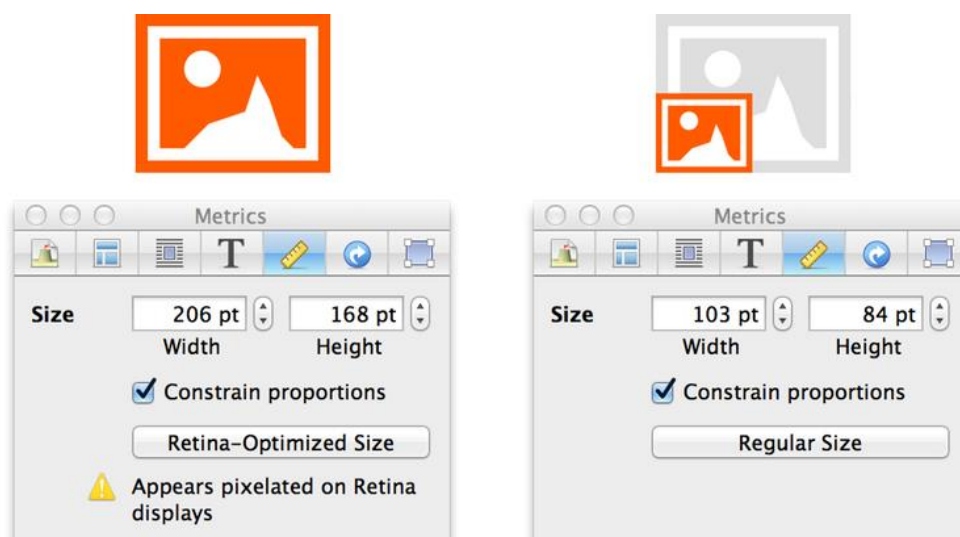


Рисунок 7. Оптимизация изображений под Retina экраны[28].

5. Полученная реализация

В данной работе была проведена оптимизация производительности сайта, реализован пользовательский интерфейс и поисковая оптимизация (SEO).

5.1 Производительность

Производительность сайта зависит от многих аспектов и есть очень много способов улучшить ее. Есть несколько простых решений, чтобы улучшить работоспособность сайта:

- Эффективный плагин кэширования
- Оптимизация изображений
- .htaccess

5.1.1 Эффективный плагин кэширования

Благодаря кэшированию пользователям не надо по несколько раз загружать одно и то же содержимое с вашей страницы, если оно не изменилось, то его обновлять не нужно, соответственно ваша страница будет загружаться последующие разы гораздо быстрее.

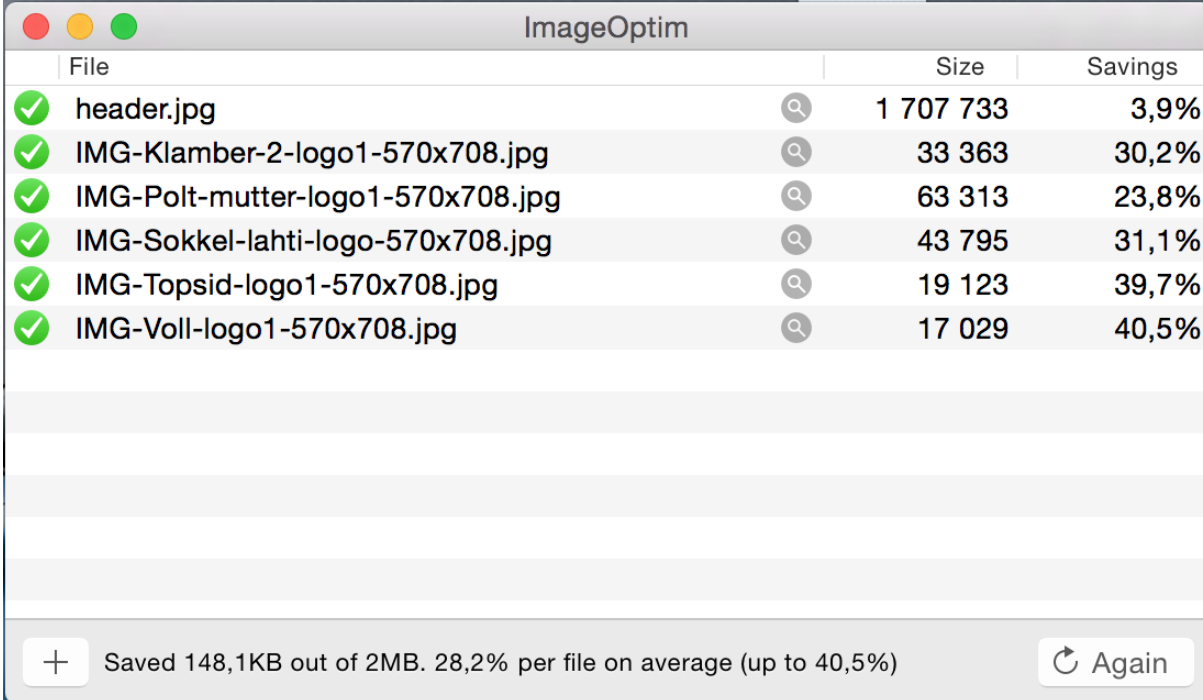
Автор использовал самый популярный плагин для кэширования W3 Total Cache. Этот плагин сложен в настройках, но оптимальные базовые опции уже выбраны по умолчанию, так чтобы и новичок смог улучшить работоспособность своей веб-страницы.

5.1.2 Оптимизация изображений

В ходе работы применялось сжатие или так называемая компрессия изображений для уменьшения занимаемой файлами памяти. При сжатии изображений качество не меняется, а размер уменьшается, как правило, на 10-30%.

Автор использовал программу ImageOptim для сжатия изображений, все изображения были дополнительно сжаты хотя бы на 20%, как это видно на рисунке номер 8. Главное изображение страницы не сжалось сильно по причине изначального сохранения его как

изображения для интернета (опция в Adobe Photoshop), где оно было оптимизировано программой.



File	Size	Savings
✓ header.jpg	1 707 733	3,9%
✓ IMG-Klamber-2-logo1-570x708.jpg	33 363	30,2%
✓ IMG-Polt-mutter-logo1-570x708.jpg	63 313	23,8%
✓ IMG-Sokkel-lahti-logo-570x708.jpg	43 795	31,1%
✓ IMG-Topsid-logo1-570x708.jpg	19 123	39,7%
✓ IMG-Voll-logo1-570x708.jpg	17 029	40,5%

+ Saved 148,1KB out of 2MB. 28,2% per file on average (up to 40,5%) ↻ Again

Рисунок 8. Оптимизация основных изображений для сайта.

5.1.3 .htaccess

.htaccess – файл дополнительной конфигурации веб-сервера Apache, этот так называемый файл настроек позволяет задавать дополнительные параметры и права для работы сервера, задавать права на доступ в определенные каталоги и т.д.

Автор выполнил следующий набор команд:

- Gzip компрессия
- Отключить ETags
- Отключить Хотлинкинг
- Заголовок контролирования кэша

Gzip компрессия файлов

Обычно на серверах отключена компрессия файлов по умолчанию, автор включил ее сам, внося этот код в .htaccess файл:

Фрагмент кода 2. Команда в .htaccess файле для активации Gzip компрессии

```
RewriteEngine On
RewriteBase /
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . /index.php [L]
```

Другой вариант, использовать W3 Total Cache, о котором говорилось ранее. В его функционал входит так же и Gzip компрессия.

ETags

ETags или Entity Tags это механизм, который используют веб-сервера и браузеры, чтобы понять совпадает ли конкретный компонент в кэше браузера клиента с этим же компонентом на сервере[29]. Entity иными словами значит компонент: скрипты, стили, изображения и т.д.

Автор решил убрать этот механизм, так как для такого маленького проекта нет никаких преимуществ в его использовании, а без него страница загружается быстрее, заголовки будут меньше. Кэширование будет происходить заголовки контролирования кэша, которые автор тоже изменил для более высокой производительности.

Фрагмент кода 3. Команда в .htaccess файле для ликвидации механизма ETags.

```
Header unset Last-Modified
Header unset ETag
FileETag none
```

Отключить Хотлинкинг

Хотлинк происходит, когда другие пользователи интернета используют изображения или другие ресурсы с вашего сервера. Это приводит к дополнительной нагрузке на вашем сервере и это, другими словами, воровство. Автор посчитал нужным предотвратить использование наших ресурсов чужими людьми, так как в нашем проекте используются уникальные изображения собственного производства.

Проблему он решил таким способом через .htaccess файл:

Фрагмент кода 4. Команда в .htaccess файле для отключения хотлинкинга.

```
RewriteEngine on
```

```
RewriteCond %{HTTP_REFERER} !^$
RewriteCond %{HTTP_REFERER} !^http://(www.)?domain_name.com/.*$ [NC]
RewriteCond %{HTTP_REFERER} !google. [NC]
RewriteCond %{HTTP_REFERER} !search?q=cache [NC]
RewriteCond %{HTTP_REFERER} !msn. [NC]
RewriteCond %{HTTP_REFERER} !yahoo. [NC]
RewriteCond %{REQUEST_URI} !^/pages/images/
RewriteRule .*.(gif|jpg|png|ico)$ - [F,L]
```

Domain_name – это домен владельца, ниже указаны поисковые движки, им разрешено делать хотлинк. Это позволит, например, видеть ваши изображения в google и других поисковиках.

Заголовок контролирования-кэша

У браузеров есть определенное время, после которого они вновь загружают статичный контент[30]. Заголовок контролирования-кэша помогает назначить время настолько долгое, чтобы браузеры пользователей не скачивали заново статичное содержимое вашей страницы.

Это значительно ускоряет работу сайта для обычных пользователей. Время указывается обычно в секундах, для всех типов изображений автор указал время равное в 30 суток, для скриптов автор указал время равное одной недели и для html кода автор указал время равное двум часам. Для скриптов и html кода указано более короткое время на случай, если будут внесены изменения, изображения практически никогда не изменяются, поэтому их можно долго хранить в кэше.

Фрагмент кода 5. Команда в .htaccess файле для контроля времени хранения кэша.

```
ExpiresActive On
ExpiresByType text/html A7200
ExpiresByType text/javascript A604800
ExpiresByType image/gif A2592000
ExpiresByType image/jpg A2592000
ExpiresByType image/jpeg A2592000
ExpiresByType image/png A2592000
ExpiresByType application/x-javascript A2592000
ExpiresByType text/css A2592000
ExpiresByType image/x-icon A2592000
```

После объединения всех команд в один файл .htaccess, как показано в фрагменте кода номер 6, автор добавил еще команды «ifModule название модуля», которая перед выполнением написанных команд, проверяет установлен ли модуль на Апаче сервере.

5.2 Пользовательский интерфейс

Благодаря фреймворку Foundation была реализована тема с адаптивным дизайном. Все изображения были оптимизированы для более быстрой загрузки, а благодаря плагину W3 Total Cache было активировано кэширование, чтобы изображения хранились в браузере посетителя после первого посещения сайта. Главная страница получилась очень простая, как видно ниже на рисунке номер 9. Навигация – это затемненная боковая панель, цель которой не привлекать внимания пользователя и сохранять внимание в центре экрана. На мобильных устройствах боковое меню скрыто, чтобы получить больше свободного пространства, которого мало.



Рисунок 9. Главная страница на разных устройствах.

Размер главной кнопки действия оставлен большим, чтобы максимально возможно заставить посетителя перейти на страницу оформления заказа. На странице одна кнопка действия (СТА)[31] большого размера, которая призывает нас нажать и пройти дальше. Так же зеленая деталь от 3D модели указывает невзначай на кнопку, тем самым

усиливая желание перейти на следующую страницу. В панели навигации снизу есть переключатель языка с возможностью смены на любой из трех языков.

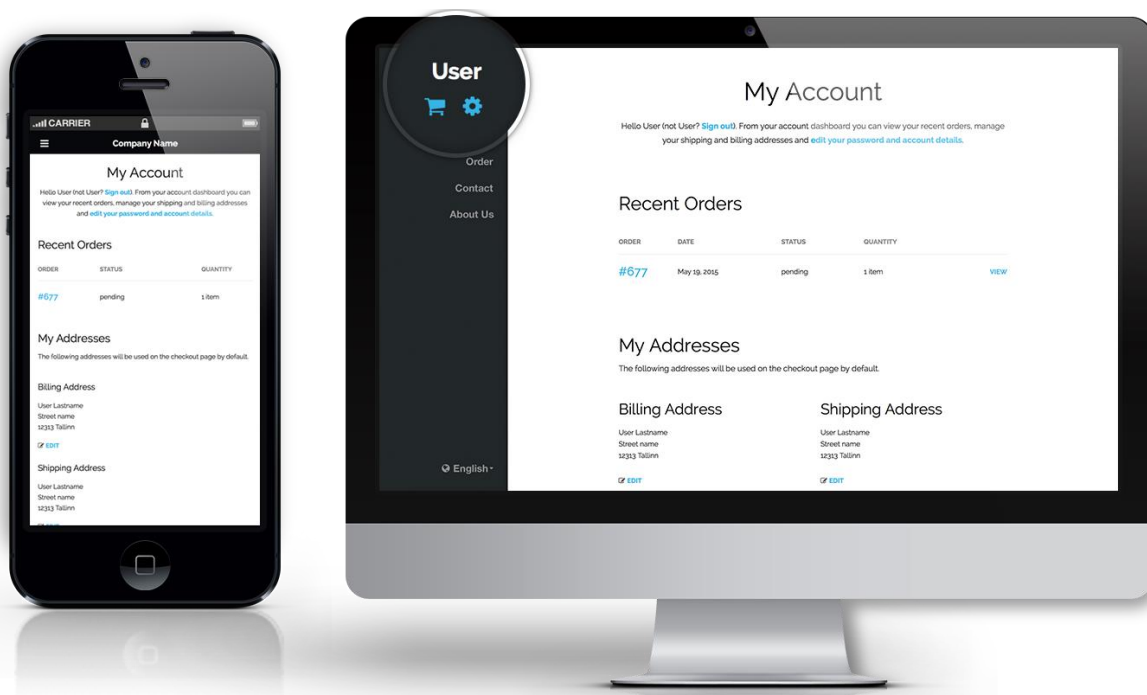


Рисунок 10. Персональная страница пользователя.

У зарегистрированного пользователя есть возможность перейти в корзину, чтобы оформить заказ или перейти на свою персональную страницу, как это изображено на рисунке номер 10. На своей личной странице пользователь может изменять свои контактные данные, просматривать недавние заказы и, при желании, выйти из аккаунта. Кроме этого, на персональной странице можно изменять свои личные данные, такие как имя, фамилия, e-майл и пароль, как это видно на рисунке номер 11. Благодаря адаптивному дизайну пользователю не составит труда поменять свои личные данные с любого устройства.

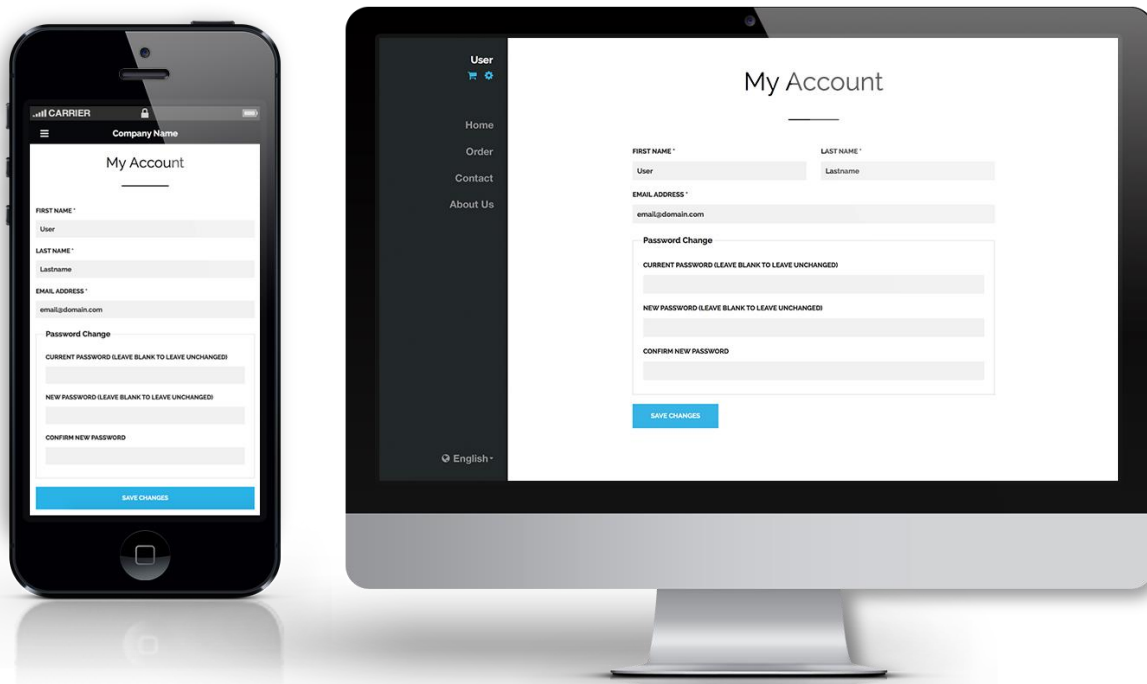


Рисунок 11. Изменение персональных данных.

Страница оформления заказа простая с максимальным использованием белого пространства, большие изображения с примерами работ, как это видно на рисунке номер 12. Также присутствует небольшое описание продукта и что представляет из себя STL модель.

Перед добавлением в корзину есть возможность добавить файл 3D модели и выбрать необходимое количество деталей, которое мы хотим получить после оплаты услуги.

3D printimise tellimine



Selleks, et alustada teie toote 3D printimisega vajame konkreetse toote **STL** -formaadis mudelit. **STL** formaadis on võimalik salvestada kasutades enamasti CAD-programme.

Laadige allolevasse lahtrisse oma **STL** mudel, valige kogus, klikkige korvi lingil ja sellega on teie tellimus meile üle antud.

STL mudel

Lisage oma STL mudel ja valige kogus

Choose File no file selected
(max file size 100 MB)

QTY 1 + - LISA KORVI

Рисунок 12. Возможность добавлять STL модель при оформлении заказа.

5.3 SEO

Поисковая оптимизация является неотъемлемой частью каждой веб-страницы, она влияет на отображение ваших страниц в поисковом движке. Чем выше ваш сайт находится в поисковом результате, тем больше посетителей.

Robots.txt это файл в корневой папке веб-страницы отвечающий за права, куда пускать поисковых роботов, а куда нет. Этот файл необходим для поисковой оптимизации.

Для реализации проекта автор написал файл robots.txt, который можно использовать для любого сайта, сделанного на основе WordPress. В него входит запрет панели управления, страницы входа, блокировка rss ленты и т.д. Код файла robots.txt находится в дополнении, фрагмент кода номер 8.

6. Заключение

Основной целью данной дипломной работы являлось создание рабочего сайта, направленного на реализацию желаний заказчика. Для достижения цели были поставлены и выполнены следующие задачи:

- Выявление функциональных и нефункциональных требований к веб-сайту
- Анализ существующих интернет движков и выбор лучшего из них
- Реализация доработок для движка
- Создание темы и дизайна
- Оптимизация готового веб-сайта

В ходе выполнения данных задач был создан многоязычный веб-сайт, с возможностью заказа услуг 3D печати и отслеживания своего заказа. Для клиентов был реализован приятный пользовательский интерфейс, для легкого и удобного использования. Благодаря дополнительным решениям была достигнута оптимальная работа ресурса.

Проделанная работа показала, что на данный момент существует множество проблем при создании веб-сайта, решить которые можно путем выбора правильных платформ и нужных дополнений к ним.

В ходе выполнения данной дипломной работы были получены практические навыки в разработке сайтов и интерфейса программ, а также навыки по оптимизации веб-страниц.

Услуги 3D печати являются сравнительно новыми, поэтому для привлечения посетителей необходимо делать простые и понятные решения, что было реализовано в данной работе.

Kokkuvõte

Selle diplomitöö peamiseks eesmärgiks oli töökorras veebilehe loomine, mis oli suunatud kliendi soovide realiseerimisele/täitmisele.

Eesmärkide saavutamiseks olid loodud ja täidetud järgmised ülesanded:

- Veebilehe funktsionaalsete ja mitte-funktsionaalsete nõuete identifitseerimine/tuvastamine
- Olemasoleva sisuhaldussüsteemi analüüs ja sellest parima valimine
- Sisuhaldussüsteemi paranduste rakendamine
- Teema ja disaini loomine
- Valmis veebilehe optimeerimine

Nende ülesannete täitmisel sai loodud mitmekeelne veebileht, koos võimalusega tellida 3D printimist ja tellimuse jälgimise teenusega. Klientide jaoks rakendati hea kasutajaliides lihtsaks ja mugavaks kasutamiseks. Tänu täiendatud lahendustele saavutati optimaalne ressursi töö.

Tehtud töö näitas, et veebilehe loomisel on hetkel palju probleeme ja neid saab lahendada valides õigeid platvorme ja nendeks vajalikke täiendusi.

Antud töö täitmise käigus saadi praktilised oskused veebilehtede ja kasutajaliidese programmide arendamise osas, samuti veebilehtede optimeerimise oskused.

3D printimise teenused on suhteliselt uued, seega klientide meelitamiseks on vaja luua lihtsad ja arusaadavad lahendused, mille võimalikkust on tõestatud antud töös.

Summary

The main purpose of this thesis was to create a working website, designed to implement customer's wishes.

To achieve the goals, following tasks were set and fulfilled:

- Identification of functional and non-functional requirements for the website
- Analysis of existing content management systems and selection of the best one
- Implementation of improvements for the content management systems
- Creating a theme and a design
- Optimization of the finished web site

In the course of performing these tasks a multilingual website with the possibility of purchasing 3D printing and tracking your order services was created.

A nice user interface for easy and convenient use for customers/clients was created. Due to additional solutions optimal performance of the resource was achieved.

Accomplished work has shown that there are a lot of problems with creating a website, which can be resolved by selecting the right platform and the right additions to them.

In the course of this thesis practical skills in web development and interface programming, as well as skills for optimizing web pages were obtained.

3D printing services are relatively new, therefore to attract visitors simple and understandable solutions have to be made, which have been implemented in this piece of work.

Использованная литература

1. **Margaret Rouse.** Web Site Definition. [В Интернете]. [Использовано: 01.05.2015].
Опубликовано на <http://searchsoa.techtarget.com/definition/Web-site>
2. **Margaret Rouse.** HTML (Hypertext Markup Language) Definition. [В Интернете].
[Использовано: 01.05.2015]. Опубликовано на
<http://searchsoa.techtarget.com/definition/HTML>
3. **Peter Coxhead.** Introduction to CSS. [В Интернете]. [Использовано: 01.05.2015].
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.136.803&rep=rep1&type=pdf>
4. **David Flanagan.** JavaScript: The Definitive Guide. [В Интернете]. [Использовано:
01.05.2015]. <https://books.google.ee/books?id=vJGlu9t9LNYC&lpg=PR13&ots=ph9bWB8J0&dq=javascript&lr&hl=ru&pg=PP1#v=onepage&q=javascript&f=false>
5. **Margaret Rouse.** Framework Definition. [В Интернете]. [Использовано:
01.05.2015]. Опубликовано на <http://whatis.techtarget.com/definition/framework>
6. **Harold Davis.** Search Engine Optimization. [В Интернете]. [Использовано:
10.05.2015].
https://books.google.com/books?hl=ru&lr=&id=EbhSiSvzQLEC&oi=fnd&pg=PA2&dq=search+engine+optimization&ots=a-NFFLLnV2&sig=-2h_RsU1q9zjCZx1U9x1d_PJVp4
7. **Margaret Rouse.** Domain Definition. [В Интернете]. [Использовано: 10.05.2015].
Опубликовано на <http://searchsoa.techtarget.com/definition/domain>
8. **Vangie Beal.** Subdomain. [В Интернете]. [Использовано: 10.05.2015].
Опубликовано на <http://www.webopedia.com/TERM/S/subdomain.html>
9. Plugins. [В Интернете]. [Использовано: 10.05.2015]. <https://wordpress.org/plugins/>
10. **Norman Walsh.** A Technical Introduction to XML. [В Интернете]. [Использовано:
01.05.2015] <http://www.xml.com/pub/a/98/10/guide0.html?page=2>
11. Пользовательский Интерфейс. [В Интернете]. [Использовано: 01.05.2015].
<http://www.rae.ru/monographs/141-4635>
12. Introduction to 3D. [В Интернете]. [Использовано: 12.05.2015].
<https://flippednormals.com/introduction-3d/>

13. **Wai Hon Wah.** Introduction to STL format. [В Интернетe]. [Использовано: 12.05.2015].
http://download.novedge.com/brands/fps/documents/introduction_to_stl_file_format.pdf
14. **Margaret Rouse.** Content management system (CMS). [В Интернетe]. [Использовано: 20.04.2015]. Опубликовано на
<http://searchsoa.techtarget.com/definition/content-management-system>
15. Introduction to HTML. Basic HTML tags. [В Интернетe]. [Использовано: 02.05.2015]. <http://network-communications.blogspot.com/2011/05/introduction-to-html-basic-html-tags.html>
16. CMS market share in the Alexa top 1M. [В Интернетe]. [Использовано: 01.05.2015]. <https://www.datanyze.com/market-share/cms>
17. Drupal CMS Benefits. [В Интернетe]. [Использовано: 01.05.2015].
<https://www.drupal.org/features>
18. Joomla Features Overview. [В Интернетe]. [Использовано: 01.05.2015].
<http://www.joomla.org/core-features.html>
19. WordPress Features. [В Интернетe]. [Использовано: 01.05.2015].
<https://wordpress.org/about/features/>
20. **Matt Cutts.** Straight from Google: What You Need to Know. [В Интернетe]. [Использовано: 01.05.2015]. <https://www.mattcutts.com/blog/seo-for-bloggers/>
21. GNU General Public License. [В Интернетe]. [Использовано: 01.05.2015].
<http://www.gnu.org/copyleft/gpl.html>
22. Foundation Zurb Features. [В Интернетe]. [Использовано: 05.05.2015].
<http://foundation.zurb.com/learn/features.html>
23. **Ethan Marcotte.** Responsive Web Design. [В Интернетe]. [Использовано: 01.05.2015]. <http://www.princeton.edu/~mlovett/reference/A%20List%20Apart-Articles-Responsive%20Web%20Design.pdf>
24. **Matthew Sonnenshein.** Google Favors Responsive Design Website. [В Интернетe]. [Использовано: 02.05.2015]. Опубликовано на
<http://www.gourmetmarketing.net/google-favors-responsive-design-website/>
25. **Steve Champeon.** Progressive Enhancement and the Future of Web Design. [В Интернетe]. [Использовано: 02.05.2015].
http://hesketh.com/publications/progressive_enhancement_and_the_future_of_web_design.html

26. **Paula Borowska.** Understanding Retina. [В Интернетe]. [Использовано: 11.05.2015]. Опубликовано на <http://designmodo.com/design-retina-displays/>
27. **Ross Miller.** MacBook Pro with Retina display review. [В Интернетe]. [Использовано: 11.05.2015]. Опубликовано на <http://www.theverge.com/2012/6/13/3082649/macbook-pro-review-retina-display-15-inch>
28. Quick Intro: Retina Displays and Graphics. [В Интернетe]. [Использовано: 11.05.2015]. <http://www.karelia.com/support/sandvox/articles/retina-displays-graphics.html>
29. Configure ETags. [В Интернетe]. [Использовано: 12.05.2015]. <https://developer.yahoo.com/performance/rules.html#etags>
30. Add an Expires or a Cache-Control Header. [В Интернетe]. [Использовано: 12.05.2015]. <https://developer.yahoo.com/performance/rules.html#expires>
31. **Jessica Meher.** 20 Critical Do's and Don'ts for Clickable Calls-to-Action. [В Интернетe]. [Использовано: 10.05.2015]. Опубликовано на <http://blog.hubspot.com/blog/tabid/6307/bid/34014/20-Critical-Do-s-and-Don-ts-for-Clickable-Calls-to-Action-SlideShare.aspx>

Дополнение

Фрагмент кода 6. Конечный файл .htaccess, написанный автором.

```
# BEGIN WordPress
<ifModule mod_rewrite.c>
RewriteEngine On
RewriteBase /
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^index\.php$ - [L]
RewriteRule . /index.php [L]
</ifModule>
# END WordPress
php_value output_buffering 1
<ifModule mod_expires.c>
ExpiresActive On
ExpiresByType text/html A7200
ExpiresByType text/javascript A604800
ExpiresByType image/gif A2592000
ExpiresByType image/jpg A2592000
ExpiresByType image/jpeg A2592000
ExpiresByType image/png A2592000
ExpiresByType application/x-javascript A2592000
ExpiresByType text/css A2592000
ExpiresByType image/x-icon A2592000
</ifModule>
<ifModule mod_headers.c>
FileETag none
</ifModule>
<ifModule mod_rewrite.c>
RewriteEngine on
RewriteCond %{HTTP_REFERER} !^$
RewriteCond %{HTTP_REFERER} !^http://(www.)?domain_name.com/.*$ [NC]
RewriteCond %{HTTP_REFERER} !google. [NC]
RewriteCond %{HTTP_REFERER} !search?q=cache [NC]
RewriteCond %{HTTP_REFERER} !msn. [NC]
RewriteCond %{HTTP_REFERER} !yahoo. [NC]
RewriteCond %{REQUEST_URI} !^/pages/images/
RewriteRule *.*(gif|jpg|png|ico)$ - [F,L]
</ifModule>
```

Фрагмент кода 7. Файл robots.txt, написанный автором.

```
User-agent: *
Disallow: /wp-admin
Disallow: /wp-includes
Disallow: /wp-content/plugins
Disallow: /wp-content/cache
Disallow: /wp-content/themes
Disallow: /index.php
Disallow: /wp-activate.php
Disallow: /wp-app.php
Disallow: /wp-blog-header.php
Disallow: /wp-comments-post.php
Disallow: /wp-config.php
Disallow: /wp-cron.php
Disallow: /wp-links-opml.php
Disallow: /wp-load.php
Disallow: /wp-login.php
Disallow: /wp-mail.php
Disallow: /wp-pass.php
Disallow: /wp-register.php
Disallow: /wp-settings.php
Disallow: /wp-signup.php
Disallow: /wp-trackback.php
Disallow: /xmlrpc.php
Disallow: /feed #RSS feed block
Disallow: */feed
Disallow: /category/*/* #Category page block
Disallow: /comments #Comments URL block
Disallow: */comment-* #Comments page block
Disallow: */trackback #Trackback URL block
Disallow: /*? #Dynamic content block
Allow: /wp-content/uploads/

Sitemap: http://www.domain_name.com/sitemap.xml
Sitemap: http://www.domain_name.com/sitemap.xml.gz
```