# **Содержание**

[Содержание 3](#_Toc169119123)

[Введение 4](#_Toc169119124)

[Глава 1. Аналитическая часть 6](#_Toc169119125)

[1.1 Анализ сферы 6](#_Toc169119126)

[1.2 Анализ web-сайтов компаний предоставляющих услуги онлайн создания дизайнов печатей 9](#_Toc169119127)

[1.3. Выбор языка разметки, языка программирования. 17](#_Toc169119128)

[1.4. Требования к интерфейсу 29](#_Toc169119129)

[1.5 Обоснование необходимости конструктора печатей и штампов 30](#_Toc169119130)

[Глава 2. Практическая часть 32](#_Toc169119131)

[2.1. Проектирование дизайна 32](#_Toc169119132)

[2.2 Верстка и программирование 36](#_Toc169119133)

[2.3. Загрузка web-страницы на хостинг и апробация 52](#_Toc169119134)

[Заключение 62](#_Toc169119135)

[Список источников 64](#_Toc169119136)

# **Введение**

Веб-конструктор печатей и штампов, именуемый в дальнейшем “Констуктор” представляет собой онлайн-сервис, который позволяет пользователям самостоятельно создавать дизайн и оформление печатей и штампов, а затем заказывать их изготовление.

Теоретическое значение конструктора заключается в том, что он отражает современные тенденции в области дизайна и производства печатных материалов. Этот сервис демонстрирует возможности автоматизации процесса создания печатей, что важно для повышения эффективности и удобства работы пользователей.

Практическое значение конструктора заключается в том, что он предоставляет пользователям возможность быстро и удобно создавать качественный дизайн печатей и штампов без необходимости обращаться к специалистам в этой области. Это экономит время и средства заказчика, а также упрощает процесс получения нужного продукта.

Актуальность конструктора обусловлена растущим спросом на персонализированные печатные материалы как среди бизнес-сектора, так и среди частных лиц. Возможность самостоятельно создавать уникальные дизайны печатей привлекает клиентов и делает такой сервис востребованным на рынке.

Целями работы конструктора являются обеспечение пользователей возможностью создания качественных и уникальных печатей и штампов, удовлетворение потребностей клиентов в персонализированных продуктах, а также повышение эффективности процесса заказа и получения готового изделия.

Объектом работы конструктора являются печати и штампы различных видов (адресные, фирменные, датеры и т.д.), а предметом работы - процесс создания дизайна и заказа этих изделий через онлайн-сервис.

Практическая значимость работы проявляется в таких аспектах как:

1. Экономия времени и ресурсов: благодаря возможности самостоятельного создания дизайна, пользователи экономят время, которое обычно уходит на обращение к специалистам. Это особенно актуально для малых и средних предприятий, где каждая минута имеет значение.
2. Повышение эффективности: использование веб-конструктора печатей позволяет быстро получить готовый дизайн и сделать заказ, минуя длительные процессы согласования и исправлений. Это способствует более оперативному выполнению задач и ускоряет процесс производства печатей.
3. Персонализация продукции: пользователи могут создавать уникальные дизайны печатей, адаптированные под свои потребности и стилистические предпочтения. Это позволяет делать продукцию более индивидуальной и соответствующей корпоративному стилю компании.
4. Доступность и удобство: конструктор 24/7, что позволяет пользователям создавать и заказывать печати в любое удобное время, без привязки к рабочему графику или местоположению. Это делает процесс заказа более гибким и удобным для клиентов.

Структура работы представляет собой веб страницу с интегрируемым в неё конструктором написаном на javascript и используя html2canvas.

# **Глава 1. Аналитическая часть**

Для анализа существующих методов решения проблемы создания веб-конструктора печатей и штампов можно провести исследование рынка и изучить существующие аналоги и решения. Ниже представлен общий анализ возможных подходов:

1. Ручной заказ печатей: Традиционный способ заказа печатей у специализированных фирм. Недостатки - длительные сроки изготовления, высокая стоимость, ограниченный выбор дизайнов, в данный момент таковые услуги предоставляет огромный спектр компаний
2. Готовые онлайн-сервисы: Существуют веб-платформы, предлагающие услуги по созданию печатей и штампов онлайн. Однако они могут быть ограничены в функционале, дизайне или возможностях настройки.
3. Программы для создания печатей: Существуют программные продукты, которые позволяют создавать дизайн печатей на компьютере. Однако требуется определенные навыки работы с графическими редакторами.
4. Использование готовых шаблонов: Можно использовать готовые шаблоны для создания печатей и штампов. Недостаток - ограниченность в индивидуальности и уникальности дизайна.

## **1.1 Анализ сферы**

Сфера создания печатей и штампов имеет значительное значение в современном мире, поскольку она играет важную роль в различных сферах деятельности, в подкрепление этого факта можно взглянуть на график популярности запроса в поисковой сети Яндекс:

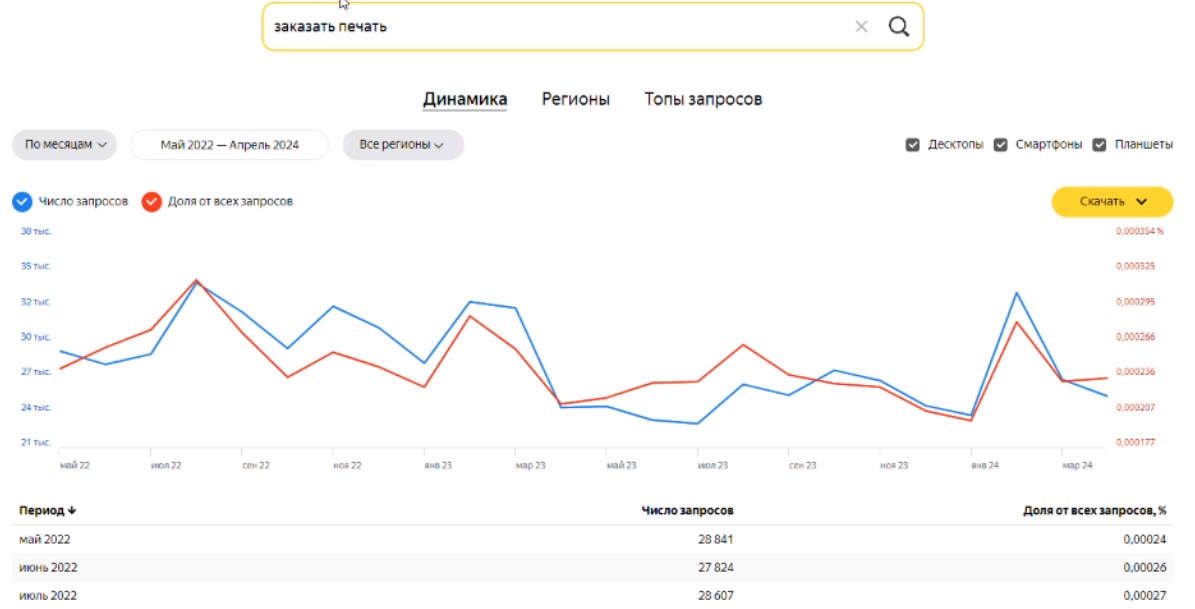


Рисунок 1 – График популярности запроса “Заказать печать”

Печати могут быть неотъемлемой частью для таких видов и сфер деятельности как:

1. Бизнес: Для многих компаний и предпринимателей печати и штампы являются неотъемлемой частью их бизнес-процессов. Они используются для оформления документов, подписей, логотипов и других важных элементов.
2. Офисная работа: В офисах и учреждениях штампы и печати используются для удобства и быстроты оформления и обработки документов. Они помогают сократить время на рутинные операции.



Рисунок 2 – Пример офисной печати

1. Личное использование: Люди также могут нуждаться в печатях и штампах для личного использования, например, для оформления почтовых отправлений, декорирования подарков или создания собственного бренда, примерами таковых является печать для сургуча или же “Сургучная”
2. Юридическая область: Использование оттиска печати при оформлении документов. Оттиск печати заверяет подлинность подписи должностного лица на документах. В юридической сфере печати и штампы являются важными атрибутами для оформления договоров, документов и других юридически значимых материалов.

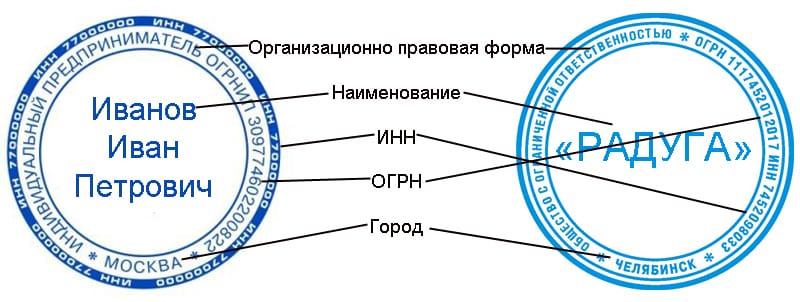


Рисунок 3 – Пример юридической печати

## **1.2 Анализ web-сайтов компаний предоставляющих услуги онлайн создания дизайнов печатей**

Для анализа интернет-сайтов конкурирующих фирм не имеет значение их территориальное местоположение. Принципы работы изготовителей печатей по сути одинаковы.

Согласно сайту 2gis в Москве находится 1406 мест где можно создать печать, по этому я решил обратить внимание на самые крупные сети филиалов подобных заведений, мною были выбраны две главенствующие компании печатного производства:

1. Копирка
2. Copy.ru



Рисунок 4 – Два магазина с самым большим количеством филиалов

“Копирка” – Российская компания специализирующаяся на полиграфических услугах, а так же услугам по дизайну, изготовлению сувенирной продукции и многом другом.

Передовой особенностью данной сети является безусловно ее удачное месторасположение, поскольку "Копирка" охватывает всю территорию города и в основном расположена рядом с главными станциями метро, что значительно облегчает доступ корпоративным и частным клиентам к широкому спектру полиграфических услуг.  
Вместе с тем, преимуществом сайта является понятный интерфейс каталога, расположенный сверху страницы.

Однако, важными недостатками сайта можно назвать его избыточное количество услуг и разнообразную информацию, что затрудняет ориентацию пользователей и поиск раздела "О компании", а также подает признаки устаревания и необходимости в доработке.



Рисунок 5 – Сайт интернет-магазина “Копирка”

На рисунке 5 изображен внешний вид веб-сайта магазина "Копирка" описывается следующим образом:  
1. Цветовая схема: Сайт выполнен в сдержанных цветах, таких как белый, серый, оранжевый или чёрный. Это создает впечатление надежности и профессионализма.  
2. Шрифт: Используется простой и удобочитаемый шрифт, который делает текст на сайте понятным и легко читаемым.  
3. Навигационная панель: На верхней части страницы расположена навигационная панель с основными разделами, такими как "Каталог", "Контакты", и огромное перечисление различных услуг компании, что несомненно играет отрицательную роль.

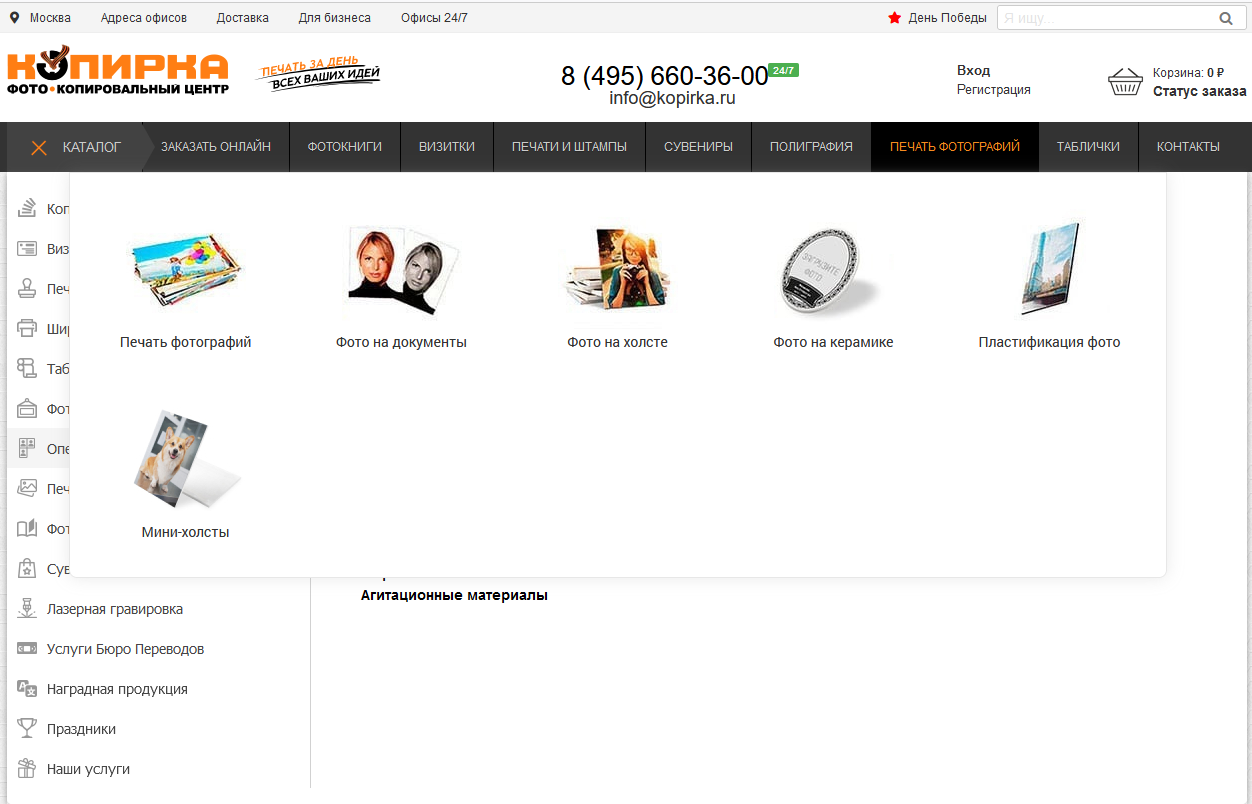


Рисунок 6 – Пример перегруженности интерфейса навигации по сайту

На шестом изображении видно, что есть избыточное количество информации на главной навигационной панели сайта Копирки. Услуги размещены как в верхнем меню, так и в выпадающем списке слева. Лучше было бы оставить их только в одном месте, чтобы избежать перегруженности информацией. Также, особенно важные услуги можно было бы выделить на главной странице, чтобы привлечь больше внимания к ним.

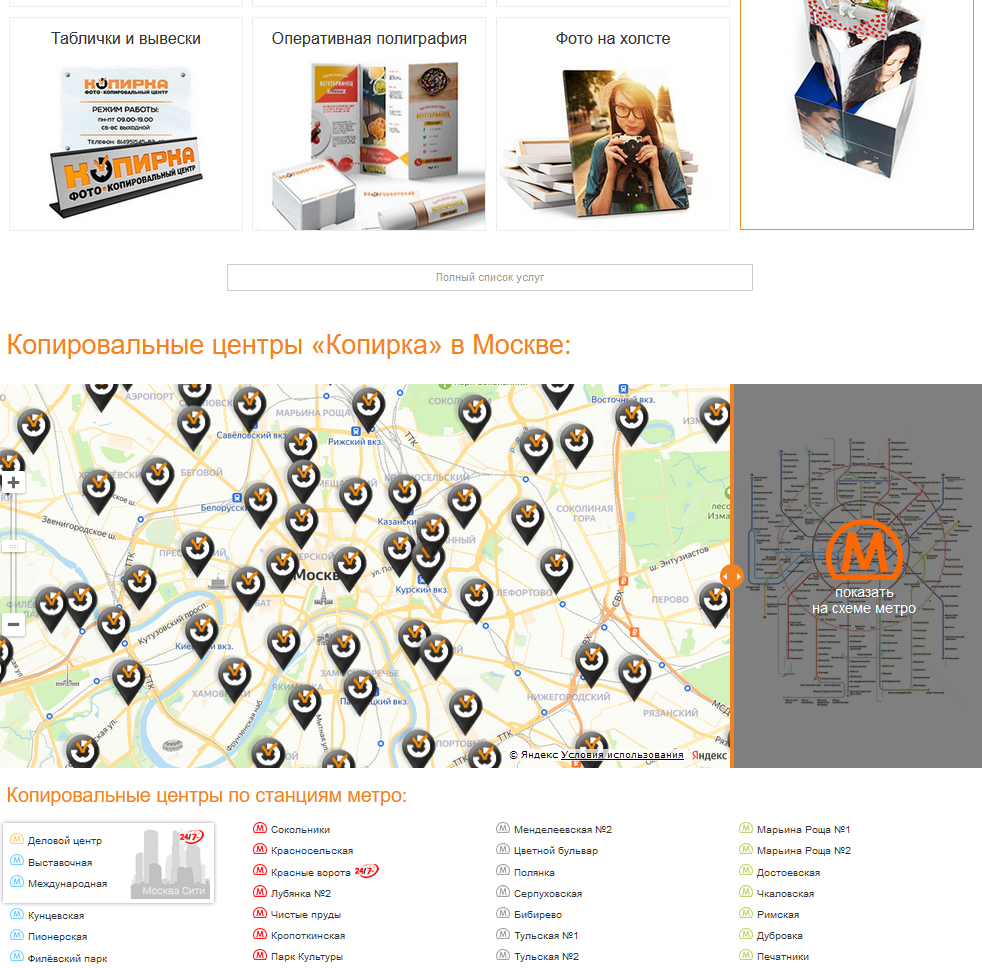


Рисунок 7 – Перегруженность главной страницы информацией

На сайте Копирки отсутствует какие либо конструкторы макетов, пользователь может лишь взять готовый оттиск печати из шаблона или заказать свой собственный у стороннего дизайнера.

Так же на сайте отсутствует конфигуратор макета, можно лишь выслать готовый или заказать из имеющихся.

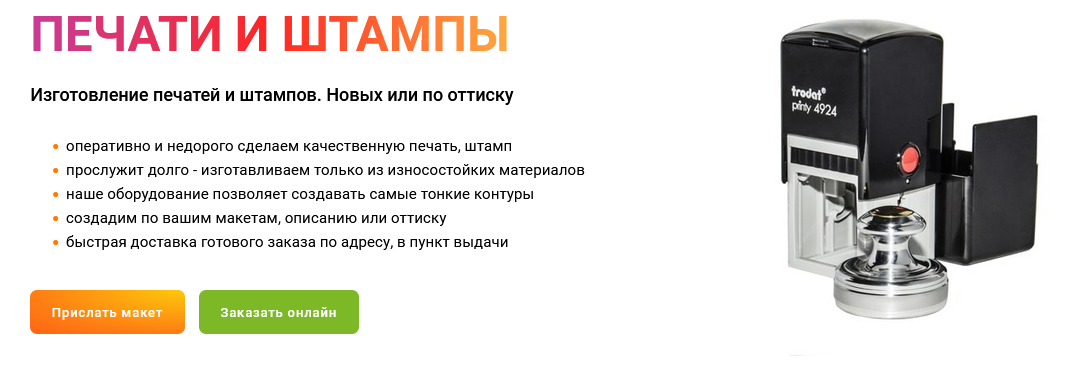


Рисунок 8 – Страница заказа печатей и штампов.

Copy.ru – федеральная сеть фотокопировальных центров. Мы умеем почти всё в полиграфии: от печати визиток и фото до широкоформатной печати и создания упаковки премиального уровня.

Достоинство этой сети является её стремительный рост, качество каждого тиража, и беспринципность в исполнении заказов, выполняют даже самые сложные.

К достоинствам сайта можно отнести его интуитивно понятный интерфейс, приятную компоновку сайта, современный дизайн, и хорошую разрозненность элементов друг от друга, что позволяет избежать моментов случайного нажатия на что-то случайно.

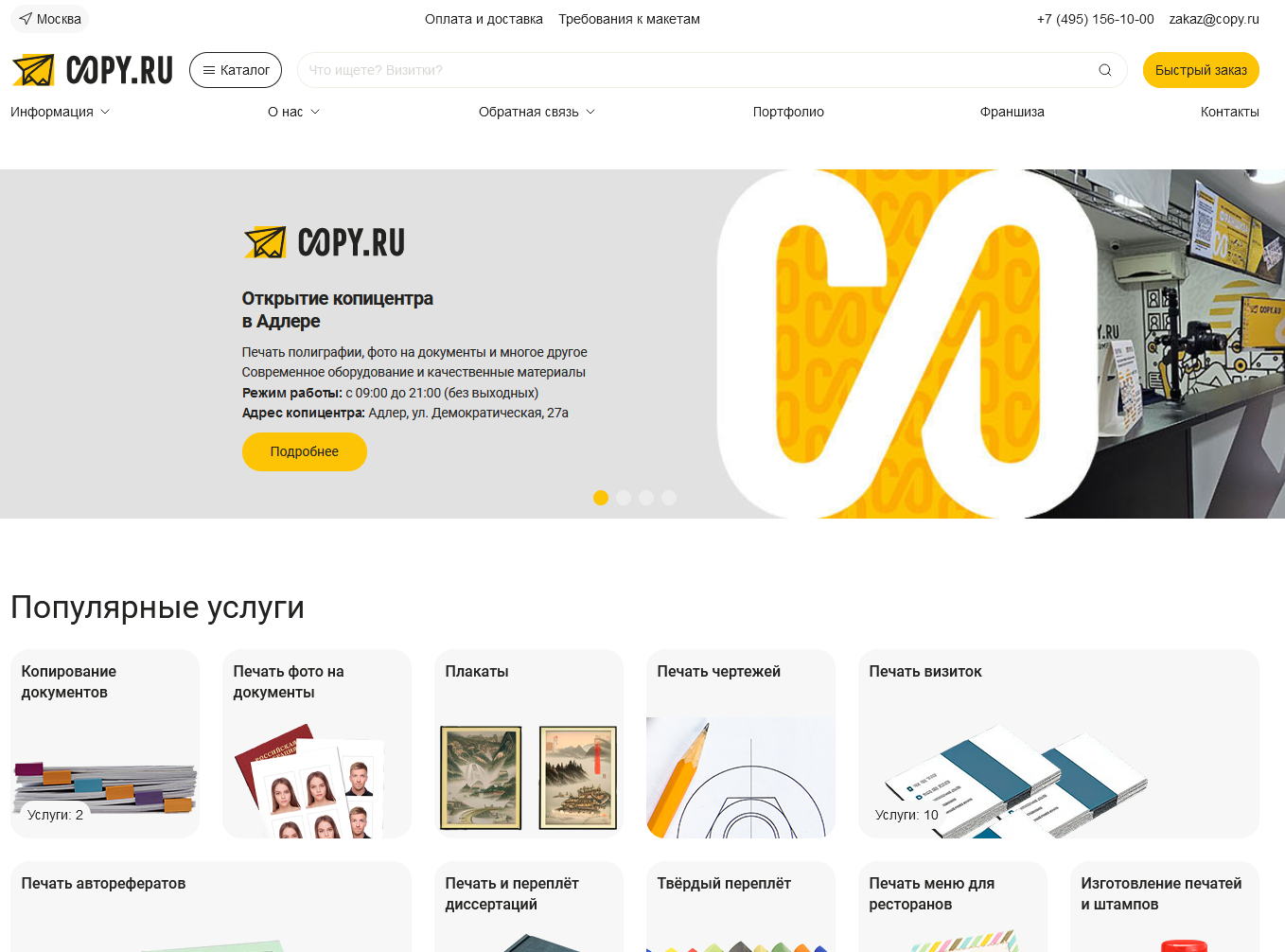


Рисунок 9 – Главная страница сайта Copy.ru

Недостатками сайта можно считать довольно неудобную форму быстрого заказа так как на ней можно довольно скудно описать нужные параметры печати если она довольно сложная, но это все перекрывается формой загрузки файла, то есть пользователь может обратится к профессиональному дизайнеру что бы тот подготовил для него дизайн печати.

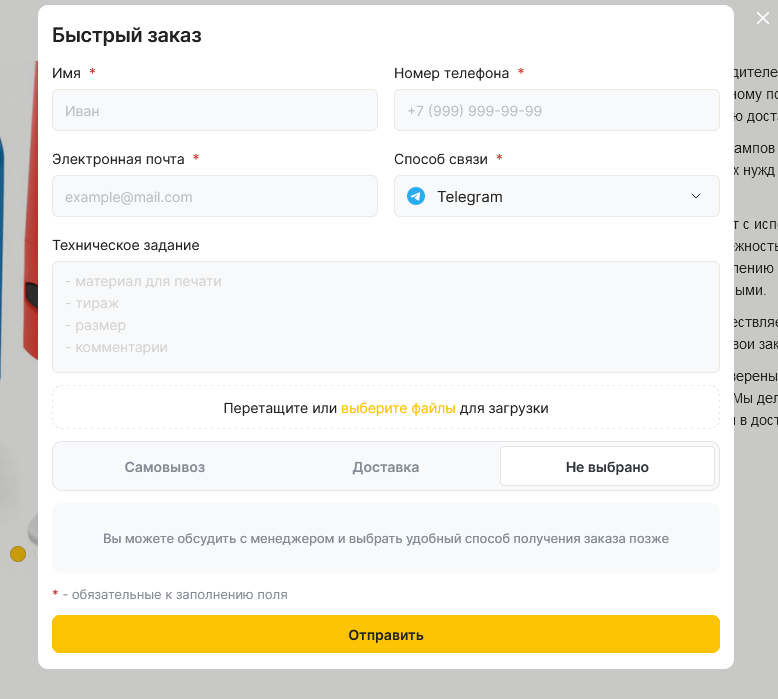


Рисунок 10 – Форма заказа на сайте Copy.ru.

Проведение анализа существующих методов позволит выявить их преимущества и недостатки, определить потребности пользователей и выработать стратегию разработки наиболее эффективного и удобного веб-конструктора печатей и штампов.

Подведем итог в сводной таблице 1.

|  |  |  |
| --- | --- | --- |
| № | Копирка | Copy.ru |
| 1.Удобство интерфейса | Нет | Да |
| 2.Направленность | Типография | Типография |
| 3.Возможность создания дизайна печати при заказе | Нет | Нет |

После тщательного анализа отрасли и рынка было принято ключевое решение о том, что разработка в области печатной деятельности представляет собой актуальное и востребованное направление для последующего развития. Причиной такого решения стала высокая необходимость в новаторских и современных решениях в данной области, при этом количество аналогичных продуктов и услуг на рынке крайне ограничено.

Это свидетельствует о большом потенциале и возможностях для успешного внедрения нового продукта или сервиса, что в свою очередь открывает перед компанией широкие перспективы роста и развития на данном рынке.

## **1.3. Выбор языка разметки, языка программирования.**

Для разработки web-конструктора применялись:

* язык гипертекстовой разметки HTML. Применяется для форматирования страничек вебсайта при отображении в браузере пользователя;
* формальный язык описания внешнего вида - каскадные таблицы стилей CSS. Применяется для стилевого дизайна страничек вебсайта при отображении в браузере пользователя;
* язык управления сценариями просмотра гипертекстовых страничек Web на стороне пользователя JavaScript;

Web-конструктор считается трудным, динамическим веб-приложением. На этот случай есть большое количество инструментов и технологий, позволяющих реализовывать такие приложения. Web-конструктор возможно воплотить в жизнь на таких языках программирования, как JavaScript, PHP, Java, Python, HTML, C, C++ и т.д. Рассмотрим некоторые из них.

**HTML**

HTML (HyperText Markup Language) – это стандартизированный язык гипертекстовой разметки, разработанный ученым из Церн Тимом Бернерсом-Ли в начале 90-х годов. Изначально HTML создавался для использования в академической среде с целью передачи между учеными и инженерами документации, научных работ и результатов исследований. Язык предлагал довольно простой набор команд, который без труда освоит каждый человек за относительно небольшое время. С помощью специальных дескрипторов «тегов» в HTML документе задаются основные объекты: поля, линии, заголовки, таблицы и прочее.

Программой, которая обрабатывает и отображает HTML документы является браузер. От его версии и возможностей зависит корректность отображения веб страниц и встроенных в них возможностей.

HTML является стандартным языком, предназначенным для создания гипертекстовых документов в среде WEB.

HTML-документы могут просматриваться различными типами WEB-браузеров. Когда документ создан с использованием HTML, WEB-браузер может интерпретировать HTML для выделения различных элементов документа и первичной их обработки. Использование HTML позволяет форматировать документы для их представления с использованием шрифтов, линий и других графических элементов на любой системе, их просматривающей.

Большинство документов имеют стандартные элементы, такие, как заголовок, параграфы или списки. Используя тэги HTML, можно обозначать данные элементы, обеспечивая WEB-браузеры минимальной информацией для отображения данных элементов, сохраняя в целом общую структуру и информационную полноту документов. Все что необходимо, чтобы прочитать HTML-документ — это WEB-браузер, который интерпретирует тэги HTML и воспроизводит на экране документ в виде, который ему придает автор.

В большинстве случаев создатель документа строго определяет внешний вид документа. В случае HTML читатель (основываясь на возможностях WEB-браузера может, в определенной степени, управлять внешним видом документа (но не его содержимым). HTML позволяет отметить, где в документе должен быть заголовок или абзац при помощи тэга HTML, а затем предоставляет WEB-браузеру интерпретировать эти тэги. Например, один WEB-браузер может распознавать тэг начала абзаца и представлять документ в нужном виде, а другой не имеет такой возможности и представляет документ в одну строку. Пользователи некоторых WEB-браузеров имеют, также, возможность настраивать размер и вид шрифта, цвет и другие параметры, влияющие на отображение документа.

HTML-тэги могут быть условно разделены на две категории:

* тэги, определяющие, как будет отображаться WEB-браузером тело документа в целом
* тэги, описывающие общие свойства документа, такие как заголовок или автор документа.

HTML-документы могут быть созданы при помощи любого текстового редактора или специализированных HTML-редакторов и конвертеров. Выбор редактора, который будет использоваться для создания HTML-документов, зависит исключительно от понятия удобства и личных пристрастий каждого автора.

В дипломном проекте с помощью HTML описываем структуру документа. HTML позволяет выделить в тексте отдельные логические части (заголовки, абзацы, списки и т.д.), поместить на Web-страницу подготовленную фотографию или картинку, организовать на странице ссылки для связи с другими документами. Это основа для сайта, HTML мы создаем структуру.

**CSS**

Несложный язык, применяемый в связке с HTML. CSS – каскадные таблицы стилей, которые необходимы для оформления изначально сухого и однообразного документа в красочных и более «живых» тонах. С помощью этого языка внешний вид веб сайтов радикально преображается – добавляется анимация, эффекты перехода, красивое форматирование текста, таблиц. Списков и прочие подобные вещи.

CSS (англ. Cascading Style Sheets – каскадные таблицы стилей) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

CSS применяется создателями web-страниц для задания цветов, шрифтов, расположения отдельных блоков и других аспектов представления внешнего вида этих web-страниц. Ведущей целью разработки CSS считалось разделение описания логической структуры web-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой web-страницы (которое теперь производится с помощью формального языка CSS). Это деление может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом.

Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером или программой чтения с экрана), или при выводе устройствами, использующими шрифт Брайля.

До возникновения CSS оформление web-страниц осуществлялось исключительно средствами HTML, непосредственно внутри содержимого документа. Однако с появлением CSS стало возможным принципиальное разделение содержания и представления документа. За счёт этого нововведения стало возможным лёгкое применение единого стиля оформления для массы схожих документов, а также быстрое изменение этого оформления.

С помощью CSS можно создавать красивый и современный дизайн веб-страниц, делая их более привлекательными и удобными для пользователей. CSS позволяет разделять структуру и содержимое веб-страницы от ее визуального оформления, что упрощает поддержку и изменение дизайна.  
  
CSS используется для применения стилей ко всем элементам веб-страницы или к определенным элементам, а также для создания анимаций, адаптивного дизайна и медиазапросов для создания отзывчивых веб-страниц, которые хорошо отображаются на различных устройствах.

Преимущества:

* Несколько дизайнов страницы для разных устройств просмотра.
* Например, на экране дизайн будет рассчитан на большую ширину, во время печати меню не будет выводиться, а на КПК и сотовом телефоне меню будет следовать за содержимым.
* Уменьшение времени загрузки страниц сайта за счет переноса правил представления данных в отдельный CSS-файл. В этом случае браузер загружает только структуру документа и данные, хранимые на странице, а представление этих данных загружается браузером только один раз.
* Простота последующего изменения дизайна. Не нужно править каждую страницу, а лишь изменить CSS-файл.
* Дополнительные возможности оформления. Например, с помощью CSS-вёрстки можно сделать блок текста, который остальной текст будет обтекать (например, для меню) или сделать так, чтобы меню было всегда видно при прокрутке страницы.

Недостатки:

* Различное отображение вёрстки в различных браузерах (особенно устаревших), которые по-разному интерпретируют одни и те же данные CSS.
* Часто встречающаяся необходимость на практике исправлять не только один CSS-файл, но и теги HTML, которые сложным и ненаглядным способом связаны с селекторами CSS, что иногда сводит на нет простоту применения единых файлов стилей и значительно удлиняет время редактирования и тестирования.
* Часто при вёрстке страниц нужно использовать одно и то же значение много раз: один и тот же цвет, один и тот же шрифт. И если это значение нужно будет изменить, то придётся менять во многих местах. В стандартном CSS нет возможностей наследования стилей, вычисляемых значений и прочих зависимостей.

Для решения этих вопросов и ускорения разработки существует несколько расширений языка CSS. Расширений в том смысле, что код CSS является валидным кодом для расширения, но не наоборот. Чтобы из кода «расширенного CSS» получился обычный CSS-файл, воспринимаемый браузером, необходимо выполнить компиляцию.

В дипломной проекте с помощью CSS формируем стиль сайта: шрифты, отступы, фон, рамки, ссылки, кнопки.

**PHP**

PHP – это язык, который имеет огромный функционал и способен полностью обеспечивать работу сложных многофункциональных сайтов. С помощью PHP создаются красочные динамичные интернет-сайты, сложные веб приложения, чаты, форумы, другие серьезные скрипты.

PHP – это серверный язык, без которого сложно представить себе нормальную работу любой серверной машины. Он используется для отладки работы Web-конструктор ов, банкингов, сложных встроенных в веб страницы скриптов. Учится не так просто, но дает своему владельцу немалые перспективы.

PHP – это широко используемый язык сценариев общего назначения с открытым исходным кодом. Говоря проще, PHP – это язык программирования, специально разработанный для написания web-приложений (сценариев), исполняющихся на web-сервере.

Аббревиатура PHP означает «Hypertext Preprocessor (Препроцессор Гипертекста)». PHP достаточно прост для изучения. Преимуществом PHP является предоставление web-разработчикам возможности быстрого создания динамически генерируемых web-страниц. Важным преимуществом языка PHP перед такими языками, как Perl и C заключается в возможности создания HTML-документов с внедренными командами PHP. Значительным отличием PHP от какого-либо кода, выполняющегося на стороне клиента, например, JavaScript, является то, что PHP-скрипты выполняются на стороне сервера.

Практический характер РНР обусловлен пятью важными характеристиками:

* традиционностью;
* простотой;
* эффективностью;
* безопасностью;
* гибкостью.

РНР предоставляет в распоряжение разработчиков и администраторов гибкие и эффективные средства безопасности, которые условно делятся на две категории: средства системного уровня и средства уровня приложения.

Из-за того, что РНР является встраиваемым (embedded) языком, он отличается исключительной гибкостью по отношению к потребностям разработчика. Хотя РНР обычно рекомендуется использовать в сочетании с HTML, он с таким же успехом интегрируется и в JavaScript, WML, XML и другие языки. Кроме того, хорошо структурированные приложения РНР легко расширяются по мере надобности (впрочем, это относится ко всем основным языкам программирования).

Нет проблем и с зависимостью от браузеров, поскольку перед отправкой клиенту сценарии РНР полностью компилируются на стороне сервера. В сущности, сценарии РНР могут передаваться любым устройствам с браузерами, включая сотовые телефоны, электронные записные книжки, пейджеры и портативные компьютеры, не говоря уже о традиционных PC. Программисты, занимающиеся вспомогательными утилитами, могут запускать РНР в режиме командной строки.

С помощью PHP происходит связь основных блоков сайта с базой данных, ее подключение к сайту. Так же происходит сортировка товаров по различной классификации.

PHP - это язык программирования общего назначения, который широко используется для создания динамических веб-сайтов и веб-приложений. Он предоставляет разработчикам мощные инструменты для работы с базами данных, обработки форм, управления сеансами пользователей и многого другого.  
PHP также позволяет интегрировать скрипты на стороне сервера с различными языками разметки, такими как HTML, JavaScript, WML и XML. Благодаря этой гибкости PHP может использоваться для создания разнообразных типов веб-приложений.  
PHP также обеспечивает высокий уровень безопасности, поскольку скрипты PHP компилируются на стороне сервера перед отправкой клиенту, что позволяет избежать проблем с зависимостью от браузеров.  
Наконец, PHP обеспечивает простую интеграцию с базами данных и обработку различных типов данных, что делает его эффективным инструментом для связи веб-приложений с базами данных и сортировки информации.

**JavaScript**

Еще один отличный многофункциональный язык. С его помощью простая неказистая страницы в интернете обрастает массой полезных и удобных функций: анимационными эффектами, кнопками навигации по странице, различными встроенными возможностями. Скрипты на JavaScript способны реагировать на определенные события, выполнять запросы на сторону сервера, работать с cookies, автоматизировать и проверять введенные в различные поля данные.

JavaScript – это язык управления сценариями просмотра гипертекстовых страничек Web на стороне пользователя. Если быть более точным, то JavaScript – это не только язык программирования на стороне клиента.

Liveware, прародитель JavaScript, является средством подстановок на стороне сервера Netscape. Однако наибольшую известность JavaScript обеспечило программирование на стороне пользователя.

Основная идея JavaScript состоит в способности изменения значений атрибутов HTML-контейнеров и свойств среды отображения в процессе просмотра HTML-страницы пользователем. При этом перезагрузки страницы не происходит. На практике это выражается в том, что можно, например, изменить цвет фона страницы или интегрированную в документ картинку, открыть новое окно или выдать предупреждение.

Название «JavaScript» является зарегистрированным товарным знаком фирмы Sun Microsystems. В данный момент JavaScript полностью занимает нишу браузерных языков. На синтаксис JavaScript оказал влияние язык Java, откуда и произошло название JavaScript; как и Java, язык JavaScript является объектным. Однако на этом их связь заканчивается: Java и JavaScript – это разные языки, ни один не является подмножеством другого.

Стандартизация языка была инициирована компанией Netscape и осуществляется ассоциацией ECMA (European Computer Manufacturers Association – Ассоциация европейских производителей компьютеров). Стандартизированная версия имеет название ECMAScript и описывается стандартом ECMA-262 (доступна в сети: на английском, на русском).

С помощью PHP происходит связь основных блоков сайта с базой данных, ее подключение к сайту. Так же происходит сортировка товаров по различной классификации.

Как и у любого язык программирования, основная задача Javascript -создавать последовательность действий, которые будут приводить к определенному результату. С помощью JavaScript создаем выпадающее меню. Так же с помощью Javascript и файла "cookie" сохраняем настройки пользователя при повторном посещении и перезагрузке страницы.

JavaScript - это язык программирования, который используется для создания интерактивных и динамических элементов на веб-страницах. Он выполняется на стороне клиента (в браузере пользователя) и позволяет изменять содержимое страницы, обрабатывать события, взаимодействовать с пользователем и многое другое без необходимости перезагрузки страницы.  
JavaScript также позволяет создавать выпадающие меню, обрабатывать файлы cookie для сохранения настроек пользователя, валидацию форм, асинхронную загрузку данных и многое другое.

JavaScript может также использоваться для создания игр, веб-приложений, анимаций, графиков, слайд-шоу, веб-сервисов и многих других интерактивных элементов. Благодаря своей популярности и распространенности, JavaScript является одним из основных языков программирования для веб-разработки и играет важную роль в создании современных интерактивных веб-сайтов.

Основная цель JavaScript - создание последовательности действий, которые приводят к определенным результатам на веб-странице. Он также предоставляет возможность взаимодействия с сервером для получения или отправки данных без перезагрузки страницы.

В целом, JavaScript является мощным инструментом для создания интерактивных веб-приложений и обеспечивает богатый пользовательский опыт на веб-страницах.

**Java**

Часто еще можно встретить web приложения на языке Java. Они обладают своими особенностями и преимуществами, среди которых называют такие:

* Возможность более легкого взаимодействия с памятью устройств;
* Способность решать нестандартные ситуации;
* Хорошие способности к фильтрации событий и информации;
* Большой набор стандартных возможностей;
* С помощью Java можно создавать функциональные сетевые приложения.

Кроме того, Java имеет обширное сообщество разработчиков, готовых помочь в решении любых проблем, а также большое количество библиотек и фреймворков, которые облегчают процесс разработки веб-приложений. Java также обладает высокой производительностью, надежностью и масштабируемостью, что делает ее отличным выбором для создания крупных и сложных веб-проектов. В целом, Java является универсальным инструментом, который обеспечивает высокую стабильность и безопасность при разработке веб-приложений.

Вот такие языки чаще всего используются для создания сайтов и для веб приложений.

HTML2Canvas - это библиотека JavaScript, которая позволяет создавать изображения (скриншоты) веб-страницы непосредственно в браузере. Это означает, что вы можете захватить изображение того, что видит пользователь на веб-странице, включая HTML-элементы, CSS-стили, изображения и другие элементы, и сохранить это изображение в формате PNG или JPEG.  
  
HTML2Canvas позволяет создавать скриншоты веб-страницы даже с содержимым, которое может быть динамически изменено с помощью JavaScript. Это полезно для создания предварительных просмотров, отчетов, областей для обрезки изображений и других сценариев, где требуется захват содержимого веб-страницы как изображения.

Библиотека HTML2Canvas представляет собой мощный инструмент, который предоставляет простой и эффективный метод создания скриншотов веб-страниц без необходимости обращения к серверу или сторонним сервисам.

Ее широкое применение в различных веб-приложениях связано с возможностью генерации изображений на основе содержимого веб-страницы, что делает ее необходимым компонентом для разработки современных и инновационных веб-приложений.  
Благодаря HTML, CSS и JavaScript, я смогу создавать красивый и удобный пользовательский интерфейс, обеспечивать визуальное оформление, а также добавлять интерактивные элементы и функциональность к приложению. PHP будет использоваться для обработки данных на стороне сервера, обеспечивая взаимодействие с базой данных и реализацию бизнес-логики приложения. А библиотека HTML2CANVAS позволит создавать скриншоты элементов HTML, что может быть полезно для сохранения изображений пользовательского контента или создания снимков экрана внутри веб-приложения. В итоге, все эти инструменты в совокупности обеспечат более эффективную разработку и повысят качество и функциональность разрабатываемого приложения.

Этот подход позволит максимально эффективно реализовать поставленные задачи и достичь желаемых результатов в сфере веб-разработки.

## **1.4. Требования к интерфейсу**

Юзабилити – степень удобства использования продукта – является одним из ключевых показателей для оценки любого digital продукта. Иными словами, юзабилити – это эргономика в цифровой среде. Хорошее юзабилити делает опыт пользователя сайта понятным, результативным и непрерывным.

Интерфейс с высоким уровнем юзабилити отличается:

* Изучаемостью. Благодаря логично расположенным элементам и знакомым паттернам взаимодействия с ними, пользователь быстро осваивает интерфейс сайта, не прилагая для этого существенных усилий;
* Эффективностью. Любой пользователь, посетивший конструктор, стремится выполнить определенную задачу. Если ему удалось достичь цели быстро и легко, то навигация сайта и организация контента отвечает требованиям юзабилити;
* Минимальным количеством ошибок. Любая ошибка нарушает непрерывность опыта пользователя и вынуждает его повторять выполненные действия снова. Полностью избавиться от ошибок нельзя, так как пользователь может сталкиваться с ними из-за собственных неверных действий. Задача UX-оптимизатора – свести к минимуму вероятность возникновения таких ошибок, а также предусмотреть их простое исправление – например, в случае с некорректно заполненной формой;
* Эстетичным дизайном. Критерий эстетики субъективен, однако внешний вид сайта должен соответствовать сложившимся нишевым стандартам.

**Выводы по первому разделу:**

При разработке конструктора главным фактором является точная формулировка цели и задачи проекта.

По результатам анализа аналогов разработки были выявлены недостатки известных интернет-магазинов по предоставлению услуг типографии, такие как:

* Перегруженность интерфейса
* Отсутствие редакторов
* Малое разнообразие оттисков
* Устаревший дизайн интерфейса
* Отстающая функциональная составляющая интернет сайтов.

При начале процесса создания интерфейса для интернет-магазинов были выявлены и учтены ключевые требования, направленные на обеспечение удобства и функциональности пользовательского опыта. На этапе проектирования интерфейса были тщательно обдуманы и размещены на сайте необходимые элементы с учетом их удобства использования.

Был создан демонстрационный макет интерфейса будущего интернет ресурса.

Важным этапом стало согласование предложенных решений с заказчиком, и только после получения его одобрения я приступил к активной разработке интерфейса будущего web конструктора печатей и штампов.

## **1.5 Обоснование необходимости конструктора печатей и штампов**

Для поддержания конкурентоспособности на современном рынке заказчиком было принято решение о разработке web-конструктора.

Если в реальном мире у организации обычно присутствует офис или магазин, то в интернете должен быть корпоративный сайт или интернет- магазин.

Цели сайтов очень разнообразны.

Некоторые сайты также могут зарабатывать на продаже продуктов или услуг прямо через свой сайт, что также является прямым источником дохода.

Кроме того, с ростом популярности и посещаемости сайта, его владельцы могут получать прибыль от спонсорских соглашений, партнерских программ, или даже создания собственного онлайн-магазина.

Объем возможных источников дохода для сайтов огромен, и многие владельцы сайтов стремятся максимизировать свои возможности в этом направлении.

Цели конструктора для предприятия ИП Горшкова Р.Ю. заключаются в следующем:

– продажа печатей и штампов а так же других услуг типографии через интернет;

– повышение спроса услугами gbr group;

– мотивация на повторный заказ услуг;

– привлечение бизнес-партнеров;

– Улучшение пользовательского интерфейса веб-сайта для удобства покупателей;

* Оптимизация процесса оформления заказа и оплаты услуг;

– Улучшение качества обслуживания и оперативность реагирования на запросы клиентов;

– Анализ конкурентов и поиск уникальных преимуществ предприятия для привлечения клиентов и партнеров;

Для достижения поставленных целей были установленны следующие задачи:

* Создание макета будущего веб приложения.
* Определение будущей функциональной составляющей.
* Создание функциональной части.
* Реализация дополнительного функционала.
* Исправление недочетов в полученном результате а так же внесение правок.

# **Глава 2. Практическая часть**

## **2.1. Проектирование дизайна**

После согласования технического задания с заказчиком, параллельно с выбором средства для создания сайта, началось проектирование макета и дизайна в графическом редакторе.

Графическим редактором для проектирования интерфейса был выбран GNU Image Manipulation Program (GIMP).

GIMP, или GNU Image Manipulation Program, - это бесплатный редактор растровой графики с открытым исходным кодом. Это кроссплатформенное программное обеспечение, которое работает в Windows, macOS и Linux. GIMP используется для различных задач по редактированию изображений, включая редактирование фотографий, ретуширование изображений и создание изображений.

Так же данная программа может использоваться в веб дизайне, при построении макетов и многого другого.

Некоторые основные функции GIMP включают в себя:  
- Ретуширование и коррекция цветов изображений.  
- Создание и редактирование слоев.  
- Использование различных инструментов рисования, кистей и фильтров.  
- Поддержка множества форматов файлов, включая JPEG, PNG, GIF, TIFF и другие.  
- Возможность расширения функциональности с помощью плагинов.

GIMP - это мощный инструмент, который можно использовать для создания высококачественных изображений. Он обладает широким набором функций, включая слои, кисти, фильтры и эффекты. GIMP также совместим с различными форматами файлов, включая JPEG, PNG и TIFF.

При помощи программы можно анализировать цвета. Инструмент гистограмма поможет фотографу проанализировать изображение в процессе обработки. Распределение яркости наглядно покажет, какая часть информации теряется при применении определенного фильтра.

Есть инструменты для изменения контраста, коррекции цветовых уровней, тонирования изображений. Эти инструменты выполняют функции фильтров**,** но позволяют использовать более тонкую настройку для каждого действия.

Изображение может быть разбито на слои. У каждого из них есть свойства: размер, прозрачность и так далее. При просмотре изображения видно несколько слоев. Подобная структура характерна для psd или xcf форматов. При редактировании изображения png или jpg оно автоматически станет первым слоем, выше которого можно добавлять новые. Можно использовать маски. Маска применяется к конкретному слою. Она может охватывать всю его поверхность или только выделенную часть.

Родным для GIMP является формат **XCF**. Он содержит всю информацию о слоях, структуре картинке и тексте на ней. Подобный подход можно встретить и в psd, родном формате программы Photoshop. Это удобно для редактирования фотографий. Текст, маски и эффекты хранятся отдельно, что позволяет редактировать одно не затрагивая другое.

После обработки фотографию стоит сохранить в одном из распространенных форматов, в противном случае ее можно будет открыть только специализированными редакторами. Программа позволяет сохранять изображение в форматах jpg, png, экспортировать файл в pdf. Ее можно использовать для конвертирования изображения. При использовании этих форматов информация о слоях теряется. Если нужно продолжить редактировать изображения позже, то копию в xcf формате не следует удалять.

GIMP  программа, которая имеет множество настроек, как для работы с фотографиями, так и для собственных окон. Можно выставить полезные инструменты в специальное меню, которое в UNIX системах может становиться отдельным окном. Помимо готовых настроек и предустановленных плагинов можно использовать сторонние скрипты и программы.

Функционал GIMP можно расширить используя дополнения, которые представляют собой программы написанные на C. Можно **добавить фильтры, кисти, рамки, градиенты или тени**. Дополнение может выполнять более сложные задачи, например копирование через слои или добавление эффектов к слою. Плагины являются полноценными программами, написанными людьми не входящими в состав разработчиков. Они могут использовать те же функции, что и пользователь или дополнять GIMP чем-то новым.

Достоинств у приложения множество:

* Бесплатная лицензия для всех;
* Есть доступ к исходному коду, можно писать дополнения и сценарии;
* Работает с большим количеством форматов;
* Доступно для Windows, macOS и Linux;
* Есть много информации по работе в открытом доступе.

Программа обладает большим количеством плюсов, но есть и недостатки:

* Нет палитры Pantone;
* Не все цветовые модели поддерживаются;
* Есть ограничения при работе с psd форматом.

Программа подойдет как новичкам, так и опытным пользователям. Она не требовательна к железу. С ее помощью можно открывать разные графические файлы, от обыкновенного jpeg до psd. Программа является мощным редактором, но при этом не введет новичка в ступор за счет простого интерфейса.

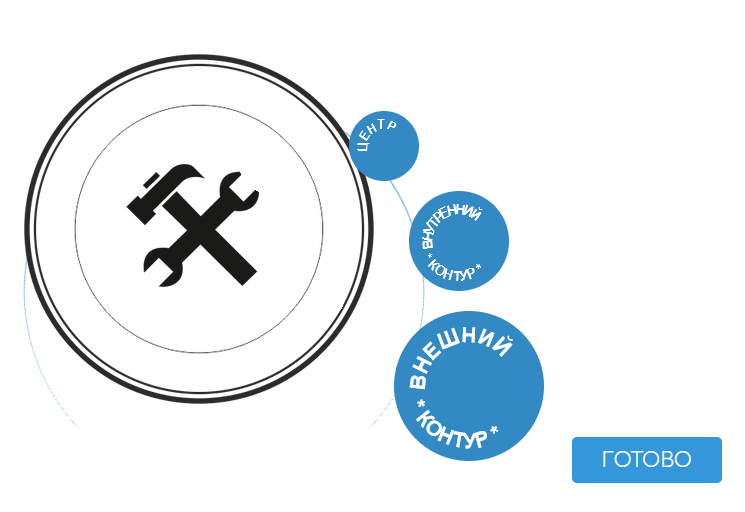


Рисунок 11 — Макет конструктора созданный в GIMP

На представленном изображении №11 изображен предварительный дизайн веб-приложения конструктора печатей и штампов, который был создан с использованием программы GIMP.

## **2.2 Верстка и программирование**

После того как внешний вид был спроектирован, все изображения найдены и шрифты определены, началась верстка (HTML5 + CSS3), подключение библиотеки и написание скриптов.

**Visual Studio Code**

Visual Studio Code (VS Code) — это редактор кода для разных языков программирования. Он относительно немного весит, гибкий и удобный. В нем можно писать, форматировать и редактировать код на разных языках [20].

VS Code не стоит путать с Visual Studio — это IDE, очень мощная и масштабная, но одновременно с этим тяжеловесная. Названия похожи, потому что у обоих средств разработки один и тот же создатель, но продукты разные по своей сути. VS Code меньше весит, проще в освоении и подходит в том числе для начинающих разработчиков.

Редактор кода существует для всех популярных операционных систем: Windows, Linux и macOS. Он бесплатный, в отличие от большинства версий «старшего брата» Visual Studio.

Кто пользуется Visual Studio Code. С VS Code работают программисты на разных языках. Например, им активно пользуются веб-разработчики, пишущие на HTML/CSS, JavaScript, PHP. Но редактор поддерживает намного большее количество языков: Python, Go, Ruby, C#, TypeScript и так далее. Он работает и с расширениями и фреймворками для популярных языков — например, с React JS и Vue.js, с языками стилей SCSS и LESS, которые дополняют CSS.

В теории пользоваться VS Code может разработчик практически на любом из современных языков. Но на практике его применяют там, где не нужны мощности полноценной IDE.

Для чего нужен VS Code. VS Code позволяет легко писать, форматировать и редактировать код на разных языках. С его помощью можно быстро создать проект и структуру файлов в нем, он подсвечивает синтаксис кода и помогает автоматически править ошибки. В нем есть возможности для отладки и запуска кода на некоторых языках.

Редактор легко расширяется, поэтому к перечисленным функциям можно добавить новые — достаточно просто скачать нужное дополнение из официального каталога. Дополнения тоже распространяются бесплатно.

Как устроен VS Code. VS Code — это расширяемый редактор кода. Он написан на фреймворке Electron, поэтому его интерфейс запрограммирован с помощью HTML, CSS и JavaScript. Интерфейс у него довольно стандартный для редакторов: рабочая область, верхнее и боковое меню.

В рабочей области открывается документ, в котором программист пишет код. Через верхнее меню можно получить доступ к разным функциям и возможностям. Через боковое меню открываются разделы контроля версий, установки дополнений, запуска и отладки кода.

В нижней части страницы можно открыть консоль. В нее будут выводиться результаты выполнения кода, разные сведения о сборке и найденных ошибках. В консоль можно писать и команды — правда, для многих из них есть горячие клавиши, а они быстрее.

Одновременно в VS Code можно открыть несколько файлов в разных вкладках. Экран рабочей области можно разделить на части, чтобы человек видел несколько файлов одновременно. Как в операционной системе, когда открыто несколько окон программ: одно справа, другое слева. Со всеми открытыми файлами можно работать.

Возможности Visual Studio Code. Написание и редактирование кода. Первое и главное назначение редактора — писать исходный код на каком-то из языков программирования, редактировать и сохранять его. Так же как редакторы текста оснащены функциями для форматирования контента, редактор кода позволяет структурировать и форматировать программы. Например, он подсвечивает синтаксис, выделяет ошибки, автоматически расставляет отступы. В нем есть и автодополнение: если ввести первые несколько букв команды, он предложит варианты, как можно ее закончить. В VS Code есть целая система для помощи в редактировании кода, ее компоненты сгруппированы под общим названием IntelliSense.

Автоматическое редактирование. Редактирование и исправление кода в VS Code можно автоматизировать. Существуют специальные плагины и расширения для редактора, которые автоматически исправляют мелкие ошибки в программе: неправильные отступы, именование, несоответствие стилю кода и так далее. Кстати, стиль кода можно задать в настройках. Конечно, сложную ошибку такой модуль не исправит: он не поймет, как это можно сделать. Но для устранения мелких недочетов возможность удобная.

Поиск, вставка и другое. Среди возможностей Visual Studio Code — быстрая навигация по документу и его редактирование. Например, строка поиска поддерживает в том числе регулярные выражения — формулы для разных текстовых сочетаний. С редактированием тоже удобно: можно написать сокращенную формулу той или иной команды, и редактор достроит ее сам. А можно, например, выделить код и закомментировать его одним сочетанием клавиш.

Горячие клавиши. Горячие клавиши позволяют применять те или иные функции, не заходя в меню. Достаточно просто нажать нужное сочетание на клавиатуре. В VS Code таких клавиш много, и их можно перенастроить под свои вкусы — как вам удобнее.

Контроль версий. VS Code «из коробки» поддерживает систему контроля версий Git. Прямо из редактора можно закоммитить изменения или откатить коммит, добавить к нему комментарий или отправить в удаленный репозиторий. Для этого не нужно входить в консоль и набирать команды: все делается через визуальный интерфейс. Сделать коммит можно в несколько кликов в специальной боковой панели.

Установка дополнений. Одна из особенностей VS Code — он очень легко расширяется. Прямо из редактора можно перейти в каталог расширений и дополнений, найти в поиске то, что вам нужно, и установить. Так подключаются поддержка других языков, плагины для автоматического исправления, конфигураторы и много чего еще. Благодаря легкой расширяемости VS Code можно полностью настроить под себя и свои нужды.

Запуск кода. Изначально редактор позволяет запускать код только для малого количества языков. Но к нему есть дополнения, которые открывают такую возможность и для других. Чтобы воспользоваться ей, нужно установить соответствующее расширение и перезагрузить VS Code. После этого в интерфейсе редактора должны появиться кнопки для сборки и запуска. Кстати, изначально в нем есть целая вкладка под названием «Запуск и отладка».

Помощь в отладке. В VS Code есть встроенный отладчик для языка JavaScript и основанных на нем технологий, таких как TypeScript. Для других языков понадобится устанавливать расширение. Но в любом случае возможность интерактивной отладки в редакторе есть, и это удобно, потому что раньше так «умели» делать только IDE. Интерактивная отладка помогает пошагово выполнять код и на каждом шаге просматривать, как изменяются данные. Так легче обнаружить и устранить ошибку.

**Преимущества Visual Studio Code.**

Простота освоения. У редактора интуитивно понятный интерфейс, его легко установить и настроить. А после этого им сразу можно начинать пользоваться. Единственной проблемой поначалу могут стать горячие клавиши и дополнительные настройки — они необычно организованы. Но со временем легко разобраться, особенно если вначале держать под рукой памятку или пользоваться туториалами.

Малый вес. По сравнению с IDE редактор быстро работает и мало весит, не так требователен к ресурсам компьютера. При этом он довольно функционален, а огромное количество расширений позволяет добавлять в него новые возможности.

Кроссплатформенность. Visual Studio Code есть для всех популярных операционных систем, так что с ним можно работать вне зависимости от ОС, которую вы предпочитаете. К тому же это позволяет быстрее и удобнее переносить проекты с одного устройства на другое.

Поддержка разных языков. Есть инструменты, которые предназначены специально для какого-то языка, например Python или C++. Visual Studio Code поддерживает огромное количество языков программирования, и они легко подключаются. Поэтому с его помощью можно вести проекты, написанные на нескольких языках одновременно.

Гибкость. У VS Code множество расширений и удобных функций. Помимо вещей, стандартных для редактора кода, вроде подсветки синтаксиса, он «умеет» много чего еще. Его можно легко и гибко настроить под себя в зависимости от потребностей разработчика и технологий, с которыми он работает.

**Недостатки Visual Studio Code.**

Ограниченная функциональность. Конечно, это зависит от того, с чем сравнивать VS Code. По сравнению с рядом других редакторов он, наоборот, более функционален. Но если сравнивать с IDE — любой редактор проигрывает, так как возможности у него намного ниже.

А для некоторых языков возможности IDE критичны. Поэтому работать в редакторе кода для разработчиков на этих языках будет не очень удобно. Например, в VS Code есть модули для Java, но Java-разработчики все же предпочитают пользоваться специализированными средами. У них больше возможностей по работе со специфическими особенностями языка вроде перевода программы в байт-код.

Медленная работа. На старых компьютерах редактор может долго запускаться и работать медленнее. Особенно это характерно для устройств, где установлен жесткий диск HDD, а не твердотельный накопитель SSD, или мало оперативной памяти.

Так происходит из-за того, что VS Code написан на Electron — это фреймворк, который позволяет писать программы для ПК с помощью HTML/CSS и JavaScript. Внутри фреймворка – целый браузер, отвечающий за отображение HTML и CSS. Соответственно, написанные на Electron программы такие же ресурсоемкие, как обычные браузеры, а значит, на слабых устройствах могут работать медленно. Это справедливо и для VS Code.

Для обеспечения более эффективного и внимательного подхода к тестированию функционала проекта, команда разработчиков приняла стратегическое решение установить инновационное бесплатное программное обеспечение Open Server. Это мощное программное средство обеспечивает возможность эмулировать полноценное окружение хостинга с виртуальными базами данных, дружественной панелью управления и широким спектром необходимых инструментов на домашнем компьютере.

Такой подход значительно упрощает и ускоряет процесс тестирования, позволяя команде разработчиков более детально проверить работоспособность приложения в условиях близких к реальным. Кроме того, использование Open Server значительно улучшает возможности по отладке кода и обнаружению потенциальных проблемных моментов в проекте, обеспечивая более гладкую и эффективную работу над устранением ошибок и повышением качества разрабатываемого программного продукта.

Для просмотра результатов разработки и отладки web-страниц я предпочел использовать бесплатный и открытый исходный код браузер Mozilla Firefox. Этот современный браузер обладает широкими возможностями для веб-разработчиков, включая инструменты для проверки кода, анализа структуры страницы, отладки JavaScript и многое другое. Благодаря своей гибкости и многофункциональности, Mozilla Firefox помогает в детальном анализе и улучшении веб-проектов, обеспечивая удобство и эффективность работы.

Так же для проверка работоспособности при других условиях были использованы два дополнительных веб браузера, это:

* Самый популярный популярный в мире в целом на данный момент браузер – Google Chrome.
* Самый популярный на данный момент браузер в России – Yandex Browser.

Главная страница сайта находится в корне проекта под названием index.html.

Важным компонентом верстки сайта являются стили, которые определяют внешний вид и оформление элементов страницы. Основные стили для сайта обычно хранятся в отдельном файле style.css, что позволяет легко управлять внешним видом сайта и обеспечить его единообразие.

На рисунке 1 в **приложении 1** можно увидеть открытый в среде разработки Visual Studio Code код html верстки главной страницы.

На рисунке 1 в **приложении 1** видно реализованный макет страницы будущего конструктора, а так же три положения для текстовых кругов, которые будут находится поверх различных окантовок печати.

**В приложении 1** на рисунке 2 изображена верстка, обозначение положения для различных элементов печати, описание внешних и внутренних контуров, положение подсказок и текста на ней, так же привязка некоторых элементов к функциям Js.

На рисунке 3 **в приложении 2** представлена возможность импорта изображений элементов печати из директории, а также демонстрируется привязка их к скриптам JavaScript и передача данных в функцию при нажатии. Этот процесс позволяет легко управлять контентом и взаимодействовать с элементами страницы.

Так же на этом рисунке вы можете заметить элементы кнопок, которые тоже привязаны к скриптам JS.

На 4 изображении в приложении 2 показан импорт различных лейблов, контуров в разметку html страницы, которые будут в дальнейшем использоваться пользователем и различными JS скриптами для подстановки нужного элемента.



Рисунок 16 — Код скрипта клиентской части написанный на JS

Функция, которая показана на рисунке 16, представляет собой инструмент, который позволяет пользователям легко и быстро получить идентификатор (ID) выбранного элемента на веб-странице. Этот ID может быть затем использован для различных целей, таких как добавление этого элемента на страницу конфигуратора печати. Таким образом, данная функция значительно упрощает процесс работы с элементами веб-страницы и помогает повысить эффективность веб-разработки и дизайна.

А так же приравнивает полученные действия к новому созданному элементу, а так же задает ему название класса для стиля css, назначает id.

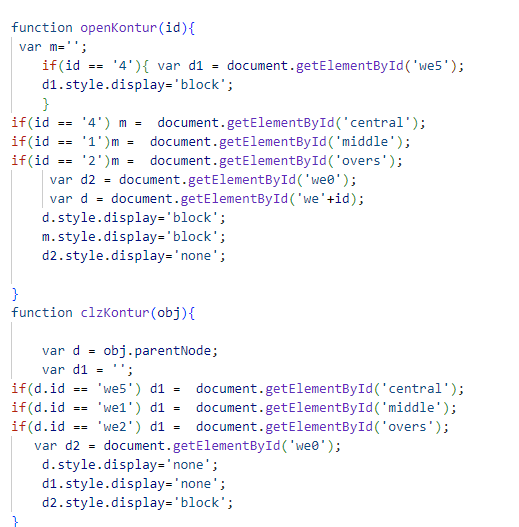


Рисунок 17 — Код скрипта клиентской части написанный на JS

Скрипт на рисунке 17 предназначен для вывода элементов печати из пред прогруженного состояния на странице в видимое и обратно, если элемент сменился его сокрытие, данный скрипт используется для различных элементов, но в основном с контурами.

Мною были использованы сторонние шрифты, именно Comfortaa.

А так же использована сторонняя библиотека для JavaScript - html2canvas.

За часть front-end (т. е. за то, что видит пользователь) отвечают скрипты, написанные на JavaScript’e.

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб- страницам.

В HTML-файле можно подключить JavaScript-файл либо прописать код в разметке между тегами <script></script> .

После выполненной верстки и программирования дизайн констуктора был выполнен полностью.

****

Рисунок 18 — Финальный вид конструктора печатей

После выбора трех требуемых элементов печати вы можете осмотреть полученный результат на странице.

Конструктор представляет собой уникальный инструмент, который позволяет выбирать из множества разнообразных контуров - как внешних, так и внутренних, каждый из которых придает изделию свой особый вид и стиль. Благодаря этой возможности, пользователь может легко настроить дизайн по своему вкусу, отключив или включив определенный контур в зависимости от желаемого эффекта. Таким образом, конструктор не только предлагает разнообразие выбора, но и открывает потрясающие возможности для творчества и самовыражения.

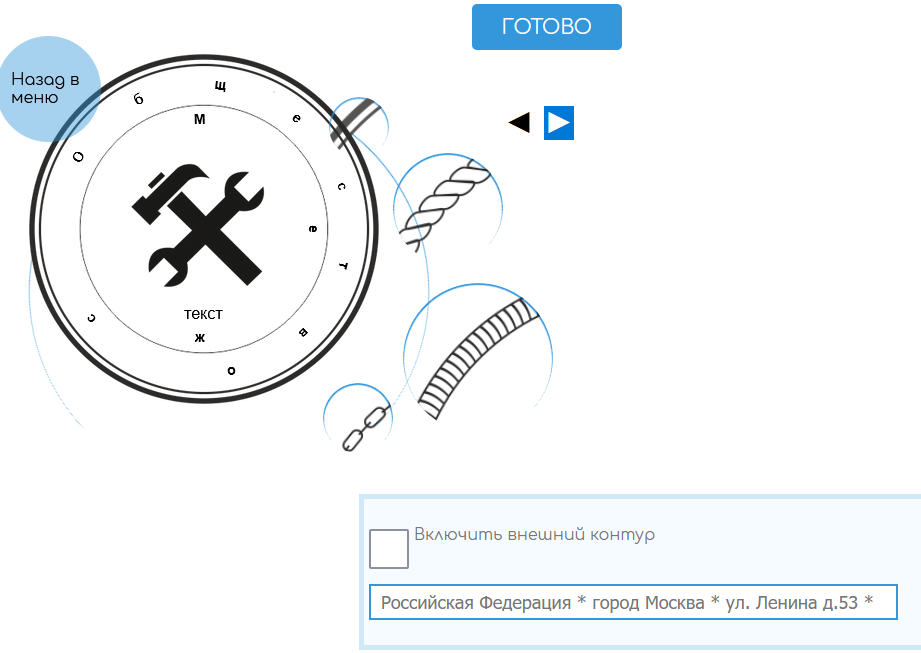
****

Рисунок 19 — Ассортимент разлиных внешних контуров

На рисунке 19 представлена огромная вариативность элементов внешнего вида контуров печати, а так же возможность его включения/выключения, и изменения текста на ней.

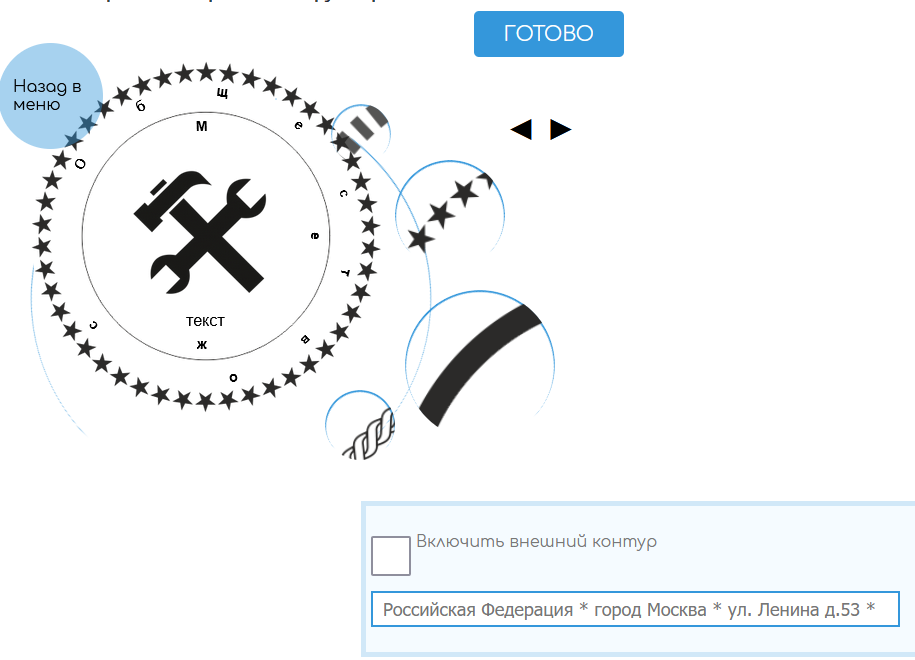
****

Рисунок 20 — Работа смены контура на деле

В процессе использования конструктора, пользователь имеет возможность ввести текст, который ему необходимо отобразить на среднем контуре, и далее выбрать самый подходящий для него паттерн.

Более того, помимо стандартных вариантов, пользователь также может воспользоваться дополнительными и даже нестандартными паттернами, доступными в переключаемом списке вариантов.

Таким образом, это позволяет пользователям украсить и оформить текст по своему вкусу, выбрав наиболее подходящий и интересный дизайн для его представления.

Благодаря широким возможностям выбора паттернов и креативной настройки текста, пользователи получают огромный диапазон инструментов для воплощения своих идей и фантазий в уникальном дизайне.

Вариативность и гибкость конструктора позволяют создавать оригинальные и запоминающиеся композиции текста, которые могут быть использованы как для декорирования личных проектов, так и для профессиональных целей.

Можно смело утверждать, что использование данного инструмента стимулирует креативность и вдохновляет на создание уникальных и стильных дизайнов, делая процесс работы более увлекательным и удовлетворительным.

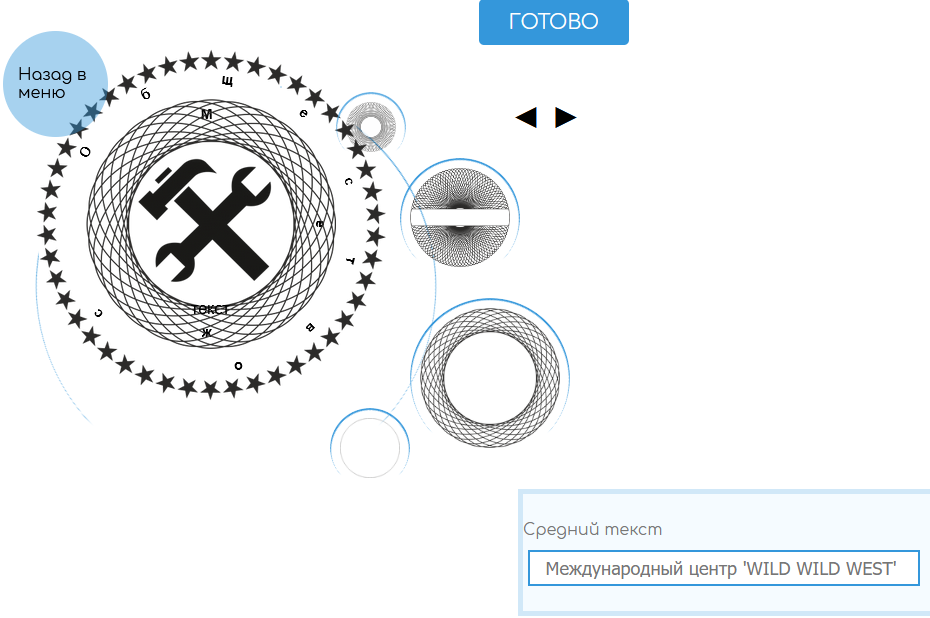


Рисунок 21 — Варианты среднего контура и поля для смены среднего текста

На данной иллюстрации под номером 21, представлена лишь одна из великого разнообразия вариантов дизайна, которые может принимать различное графическое изображение.

В данном конкретном случае, специфический вариант исполнения печатного материала отличается от стандартных шаблонов и предлагает необычные подходы к созданию внешнего вида и внутреннего оформления изделия.

Бросается в глаза необычность и смелость выбранных решений в области оформления краев и границ печати, что придает продукту уникальный и запоминающийся внешний вид.

Каждый элемент дизайна, включая окантовку, является важным фактором, определяющим общее впечатление от продукта и способствующим формированию его собственного стиля и узнаваемого облика среди других аналогичных дизайнов.

Так же человек может поменять логотип на предлагаемый или загрузить свой, который может быть создан профессиональным дизайнером, но вместе с логотипом человеку так же будет предложено придумать и записать текст под ним.

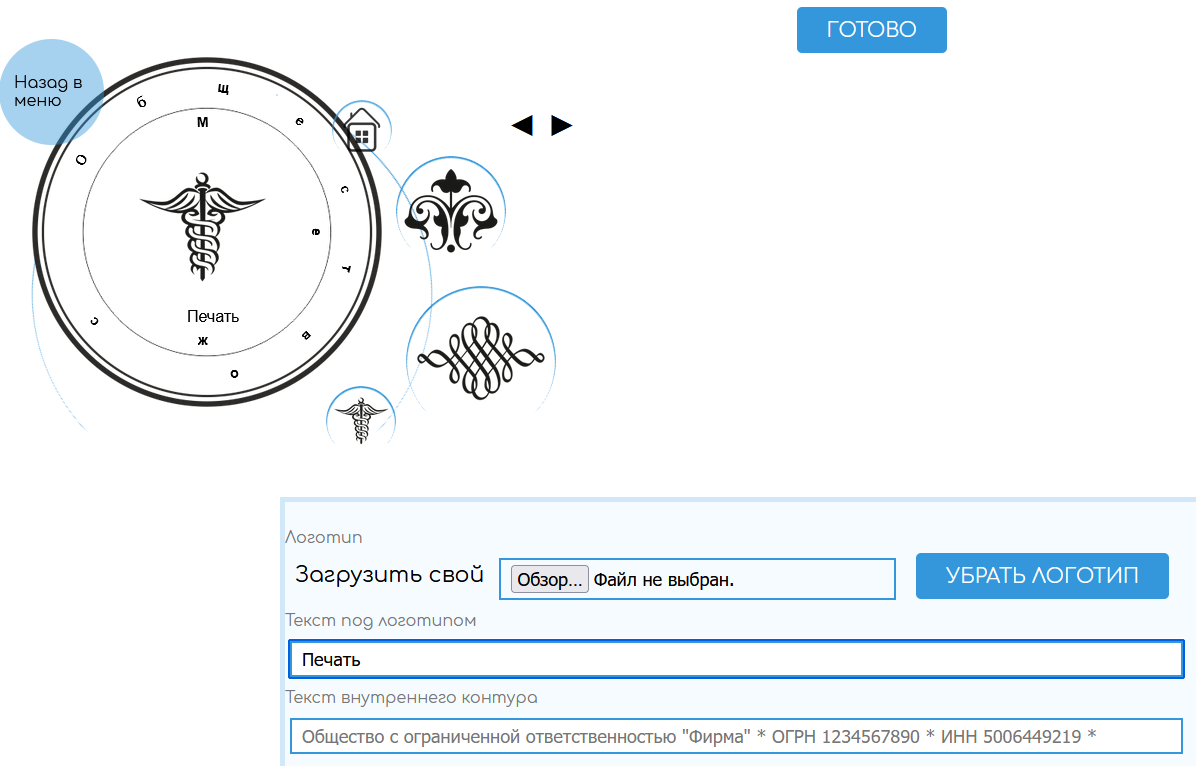


Рисунок 22 — Варианты центрального логотипа и возможности смены текста

Различные элементы печати могут быть передвинуты и аранжированы хаотичным образом как изображено на рисунке 22 в соответствии с пожеланиями и предпочтениями конечного потребителя. Гибкость в дизайне позволяет адаптировать продукт под индивидуальные потребности и вкусы заказчика, создавая уникальный продукт, который идеально соответствует его ожиданиям и желаниям. Удаление или замена логотипа также является одним из способов персонализации продукта, делая его более индивидуальным и соответствующим конкретному бренду или потребителю. Главное, чтобы такие изменения были оправданы и не ущемляли целостность и цель продукта, сохраняя его эстетическое и функциональное качество.

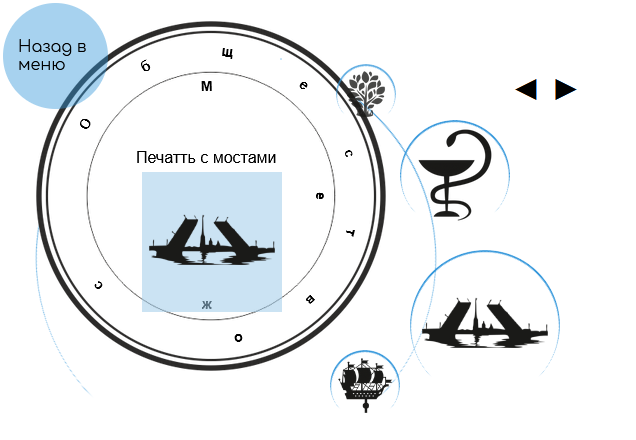


Рисунок 24 — Передвинутые мосты.

После завершения всех необходимых манипуляций с макетом печати, вы имеете возможность сохранить его в различных форматах, включая JPG и PNG. Такое сохранение позволит вам сохранить изображение в высоком качестве и с различными настройками сжатия, что обеспечит оптимальное соотношение качества и размера файла. Формат JPG, хотя и является постоянным форматом сжатия, отлично подходит для фотографий и изображений с плавными переходами цветов. С другой стороны, формат PNG, не предлагающий потерь в качестве, подходит для изображений с прозрачными фонами и лучше сохраняет детали и цвета. Таким образом, выбирая между этими форматами, вам стоит учитывать особенности вашего макета и требования к итоговому файлу.

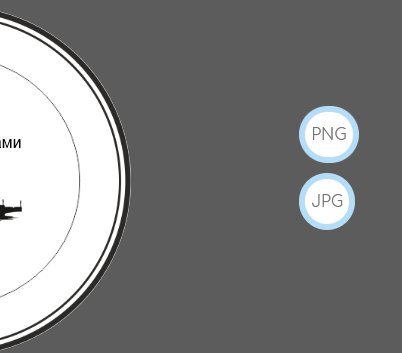


Рисунок 25 — Сохранение макета.

Рисунок 25 показывает нам что макет печати может быть сохранен в нескольких форматах изображений. Таковыми являются JPG и PNG.

Протяжении продолжительного времени данная функция работала нестабильно, приводя к ситуации, когда шрифт на сохраненном изображении смещался влево. Но ошибка была устранена, и в настоящее время приложение функционирует без проблем.

Преимущество данного расположения элементов в том что оно максимально просто и интуитивно понятно для человека нового в компьютерной сфере, и в дизайнерских делах в целом.

Так же при нажатии кнопки сохранения конструктор затемняется чтобы пользователю было лучше видно всплывающие кнопки и финальный вариант готовой печати.

## **2.3. Загрузка web-страницы на хостинг и апробация**

Для проведения тестирования конструктора в условиях, максимально приближенных к реальным, было принято решение загрузить сайт на хостинг. После проведения исследования различных вариантов, мной был выбран хостинг от компании Beget.

Этот выбор был обусловлен тем, что Beget является международным хостинг-провайдером, который аккредитован как национальными, так и международными регистраторами доменных имен.

Кроме того, Beget предоставляет широкий спектр услуг, включая виртуальный хостинг, аренду виртуальных серверов, аренду выделенных серверов и регистрацию доменных имен.

Благодаря этому, возможности тестирования на хостинге от Beget стали более всеобъемлющими и позволили более точно оценить работу конструктора на реальном хостинге.

Преимуществами данного хостинга является:

– Гибкая система настройки под себя свой хостинг, удобное меню.

– Быстрая, оперативная работа технической поддержки.

– Вежливая обратная связь которая помогает разобраться в вопросах которые даже не в их компетенации, на которые они должны давать ответы.  
– Очень удобная панель,легко управлять множеством аккаунтов.

– Решают даже то что не входит в рамки их поддержки.

Тестирование сайта прошло успешно, все модули работают без нареканий.

Есть возможность выгрузить страницу или интегрировать конструктор с этой на какую либо другую с сайта.

**** Рисунок 26 — Сайт с конструктором.

На представленном в 26 рисунке можно увидеть завершенный результат продукта, размещенного на веб-хостинге. Визуальное отображение продукта позволяет нам увидеть и оценить его окончательный вид после завершения работы с конструктором. Важно отметить, что весь процесс прошел успешно и без каких-либо проблем или помех.

Работа с конструктором шла гладко и эффективно, что позволило достичь желаемого результата и создать удивительный продукт, который теперь доступен на веб-хостинге для всех пользователей.

Так же во время загрузки возникли небольшие технические неполадки, а именно с выгрузкой изображений с хостинга, но данные проблемы были успешно исправлены а приложение конструктора загружено на сервер.

**ГЛАВА 3. ЭКОНОМИЧЕСКИЕ АСПЕКТЫ РАЗРАБОТКИ КОНСТРУКТОРА**

**3.1. Денежные и временные затраты**

В настоящем разделе дипломного проекта производится расчет затрат на разработку конструктора по построению индивидуального дизайна элементов полиграфии для магазина печатей и штампов для ИП Горшков Р.Ю.

Целью данного раздела является расчет:

⦁ себестоимости проекта

⦁ экономической эффективности проекта.

Для подсчета себестоимости проекта и экономической эффективности данного программного продукта, нужно знать следующие составляющие:

⦁ расчет затрат на энергоресурсы;

⦁ амортизационные отчисления;

⦁ расчет фонда заработной платы;

⦁ прочие (накладные) расходы.

**3.2. Исходные данные для расчетов**

**Таблица 2 – Исходные данные**

|  |  |  |
| --- | --- | --- |
| **Показатели** | **Единицы измерения** | **Значения** |
| **Страховые взносы (от заработной платы с учетом районного коэффициента), в том числе:** | **%** | **30** |
| **- отчисления в пенсионный фонд** | **%** | **22** |
| **-отчисления в фонд социального страхования** | **%** | **2,9** |
| **- отчисления в Федеральный и**  **Территориальный фонд обязательного медицинского страхования** | **%** | **5,1** |
| **Заработная плата специалиста (техника-программиста)** | **руб./мес.** | 102 500 |
| **Фонд рабочего времени в 2024 году**  **при 40-часовой рабочей неделе\*** | **дней**  **часов** | **1979** |
| **Стоимость компьютера** | **руб.** | **90 000** |
| **Срок полезного использования компьютера** | **лет** | **5** |
| **Амортизационные отчисления 100%:5 лет** | **%** | **20** |
| **Потребляемая мощность компьютера (в час)** | **кВт** | 0,108 |
| **Тариф на электроэнергию** | **руб./ кВт-час.** | 6,73 |
| **Затраты на текущий и профилактический ремонт компьютера\*\* (90 000 \* 20 % / 12), включая услуги Интернет-провайдера** | **руб** | **2400** |
| **Затраты на расходные материалы (см. Таблица 2)** | **руб** | **5 500** |

\* Всего в 2024 году будет 365 дней, из которых 247 рабочих и 118 выходных. Рабочее время при 40-часовой неделе составит 1 979 часов, при 36-часовой неделе — 1 773,4, а при 24-часовой неделе — 1 180,6.

\*\*Затраты = Стоимость OC \* Норму амортизации / 12 месяцев.

Таблица 3. Затраты на расходные материалы

|  |  |  |
| --- | --- | --- |
| Задача | Затраты | Ресурс |
| Разработка макета | 5000 | Веб-дизайнер |
| Выбор шаблона | 500 | Веб-дизайнер |
| Создание сайта | 5 500 |  |

Для расчета стоимости проделанной работы потребуется узнать:

⦁ Среднюю заработную плату программиста на предприятии по г. Москва;

⦁ Затраты на оборудование;

⦁ Затраты на электроэнергию.

Среднее значение заработной платы у техника программиста необходимо вычислить при помощи анализа заработной платы с нескольких предприятий по г. Москва (Московская область).

Ниже представлены 4 примера вакансий:

* Заработная плата в размере 120 000 рублей

Источник: <https://hh.ru/vacancy/100861733?query=Web+программист&hhtmFrom=vacancy_search_list&customDomain=1>

* Заработная плата в размере 90 000 рублей

Источник: <https://hh.ru/vacancy/98792075?query=web-программист&hhtmFrom=vacancy_search_list>

* Заработная плата в размере 90 000 рублей

Источник: <https://hh.ru/vacancy/99735471?query=web-программист&hhtmFrom=vacancy_search_list>

* Заработная плата в размере 110 000 рублей

Источник: <https://hh.ru/vacancy/100310764?query=web-программист&hhtmFrom=vacancy_search_list>

Далее высчитываем среднее значение:

(120000 + 90000 + 90000 + 110000)/4 = 102500 рублей, средняя зарплата у техника программиста.

Вычисляем почасовую оплату. Для этого нужно разделить среднюю заработную плату за год на фонд рабочего времени в 2024 году:

(102500 \* 12 месяцев)/1979 = 621 рублей (среднегодовая почасовая оплата)

На создание, предоставленной в дипломном проекте, web-сайта “Дезинсектор 3” в общем объеме ушло 70 часов.

621 \* 70 = 43 470 рублей - оплата всех часов работы.

Общие затраты на оплаты труда составили:

Заработная плата + Страховые взносы во внебюджетные фонды (30 %):

43 470 + 43 470 \* 30% = 56 511 рублей.

Затраты на электроэнергию вычислим, узнав:

⦁ Тариф на электроэнергию;

⦁ Рассчитать стоимость.

Тариф на 1 кВт/ч = 6,73 рубля.

Источник: https://mosenergosbyt.info/tarify/

Примерное использование кВт в час составляет 0,108. Получается на электроэнергию у нас уходит: 0,108 \* 6,73 = 0,726 рубля.

В месяц при 8-часовом рабочем дне:

0,726 \* 70 = 50 рублей – затраты на электроэнергию.

Таким образом, себестоимость разработки web-конструктора для создания печатей и штампов ИП Горшков Р.Ю. составила 64 461рубля.

Прибыль составит 12 892, цена разработки - 77 353 рублей.

Розничная цена составит 92 823 рублей с учетом НДС 20%.

|  |  |  |
| --- | --- | --- |
| Показатели | Условное обозначение | Значение |
| 1. Прямые расходы, в том числе: |  |  |
| 1.1 Затраты на оплату труда | ЗП | 43 470 |
| 1.2 Страховые взносы во внебюджетные фонды | СтрВзн | 13 041 |
| 1.3. Затраты на текущий и профилактический ремонт (амортизационные отчисления (90000\*20%/12) + услуги Интернет провайдера) | Зпр.р | 2400 |
| 1.4. Затраты на материалы, необходимые для эксплуатации | Зматер | 5 500 |
| 1.5. Затраты на электроэнергию | Зэл | 50 |
| 2. Итого затраты на разработку программного продукта (полная себестоимость) | ОбщЗ | 64 461 |
| 3. Плановая прибыль (20%) | П | 12 892 |
| 4. Цена РАЗРАБОТЧИКА | Ц | 77 353 |
| 5. НДС (20% от цены) | НДС | 15 470 |
| 6. Цена продажная в розницу с НДС | РознЦ | 92 823 |

**3.3 Экономический эффект и эффективность**

Стоимость поставки 92 834 рублей продукта для заказчика сопоставима с предложениями других поставщиков.

Ввиду проектно-исследовательского характера работы и отсутствия конкретных объектов внедрения приведены общие данные экономического эффекта внедрения разработки веб конструктора печатей и штампов для ИП Горшков Р.Ю.

Ожидаемый экономический эффект от внедрения разработки веб-конструктора для ИП Горшков Р.Ю.:

- повышение прибыли, благодаря:

– Увеличению клиентской базы.

– Ускорению работы с клиентами.

– Уменьшению затрат на дизайнерские услуги.

- повышение управляемости компании, благодаря:

– Своевременной реакции на невыполнение плана.

- уменьшение затрат и потерь компании, благодаря:

– Уменьшению влияния человеческого фактора на конечный результат.

– Повышению уверенности в финальном дизайне печати.

– Уменьшению количества брака.

Заключение

В результате выполнения дипломной работы был разработан и реализован конструктор для создания макетов печатей. Для ИП Горшкова Р.Ю. В процессе работы были практически использованы различные методы проектирования, прототипирования и создания визуальной составляющей web-приложений, а также реализации функционала, учтены требования заказчиков и проведено тестирование разработанного решения.

Работа позволяет упростить и ускорить процесс создания печатей, а также повысить их качество и точность. Предложенный конструктор отличается удобным интерфейсом, а также возможностью индивидуальной настройки и адаптации под разные потребности пользователей.

Полученный результат расширяет возможности для рядовых пользователей и офисных работников, обеспечивая им удобный и эффективный инструмент для создания уникальных макетов печатей.

Данный конструктор для создания макетов печатей не только обеспечивает пользователей удобным и эффективным инструментом для работы, но также способствует сокращению времени на разработку макетов и повышению производительности. Благодаря возможности индивидуальной настройки и адаптации, пользователи могут создавать уникальные и качественные макеты, отвечающие их требованиям.

Кроме того, разработанный решение имеет потенциал для дальнейшего расширения функционала и добавления новых возможностей, что делает его перспективным инструментом для профессионалов в области дизайна и бизнеса. Полученный результат оказывает значительное влияние на работу пользователей, облегчая им процесс создания макетов и повышая качество их продукции.

Так же данный макет позволяет создавать уникальные и креативные дизайны, макеты печатей которые помогут выделится на рынке и привлечь дополнительное внимание клиентов.

Это инструмент для благоприятного развития имиджа компании или бренда, что очень важно в условиях современной динамичной и оперативной бизнес среды.

Список использованных источников информации

**Книжные (Печатные) издания:**

1. Скит Джон: JavaScript для профессионалов. Тонкости программирования, 2020. — 218 с.
2. Макнейл, П. Веб-дизайн. Книга идей веб-разработчика /П.Макнейл – СПб.: «Питер», 2021. – 288 с.
3. Ясенев, В.Н. Информационные системы и технологии: Учебное пособие / В.Н. Ясенев. - М.: ЮНИТИ, 2022. - 560 c.
4. Астелс, Дэвид; Миллер Гранвилл; Новак, Мирослав, Практическое руководство по экстремальному программированию, Пер. с англ. - М.: Издательский дом "Вильямс", 2020. — 320 с.
5. Познакомьтесь, JavaScript – Кайл Симпсон 2021г.
6. Область видимости и замыкания – Кайл Симпсон 2021г.
7. Замыкания и объекты – Кайл Симпсон 2021г.
8. Типы и грамматические конструкции – Кайл Симпсон 2021г.
9. Асинхронная обработка и оптимизация – Кайл Симпсон 2021г.
10. ES6 и не только – Кайл Симпсон 2021г.
11. Джон Дакетт: Javascript и jQuery. Интерактивная веб-разработка 2020г.
12. Резиг, Фергюсон, Пакстон: JavaScript для профессионалов – 2020г.
13. Валерий Янцев: JavaScript и PHP. Content management system 2023г.
14. Хантер, Инглиш: Многопоточный JavaScript 2022г.
15. Хантер, Инглиш: Многопоточный JavaScript 2022г.
16. Скотт, Пауэрс, Макдоналд: JavaScript. Рецепты для разработчиков 2023г.
17. Андрей Диков: Web-программирование на JavaScript. Учебное пособие для СПО 2022г.
18. Валерий Янцев: JavaScript. Готовые программы. СПО 2021г.
19. Сергей Беляев: Разработка игр на языке JavaScript. Учебное пособие 2023г.
20. **Интернет-источники**
21. Справочник по HTML разметке: [Электронный ресурс]. – Режим доступа: свободный. URL: https://www.w3schools.com/tags/ (дата обращения 06.04.2024)
22. Справочник по разметке CSS: [Электронный ресурс]. – Режим доступа: свободный. URL: https://www.w3schools.com/css/ (дата обращения 07.04.2024)Fh
23. Туториал по JS: [Электронный ресурс]. – Режим доступа: свободный. URL: https://www.w3schools.com/js (дата обращения 08.04.2024)
24. Документация по “Visual Studio Code”: [Электронный ресурс]. – Режим доступа: свободный. URL: https://code.visualstudio.com/docs (дата обращения 06.04.2024)**Приложения**

*Приложение 1*

Рисунок 1 – Код html верстки главной страницы

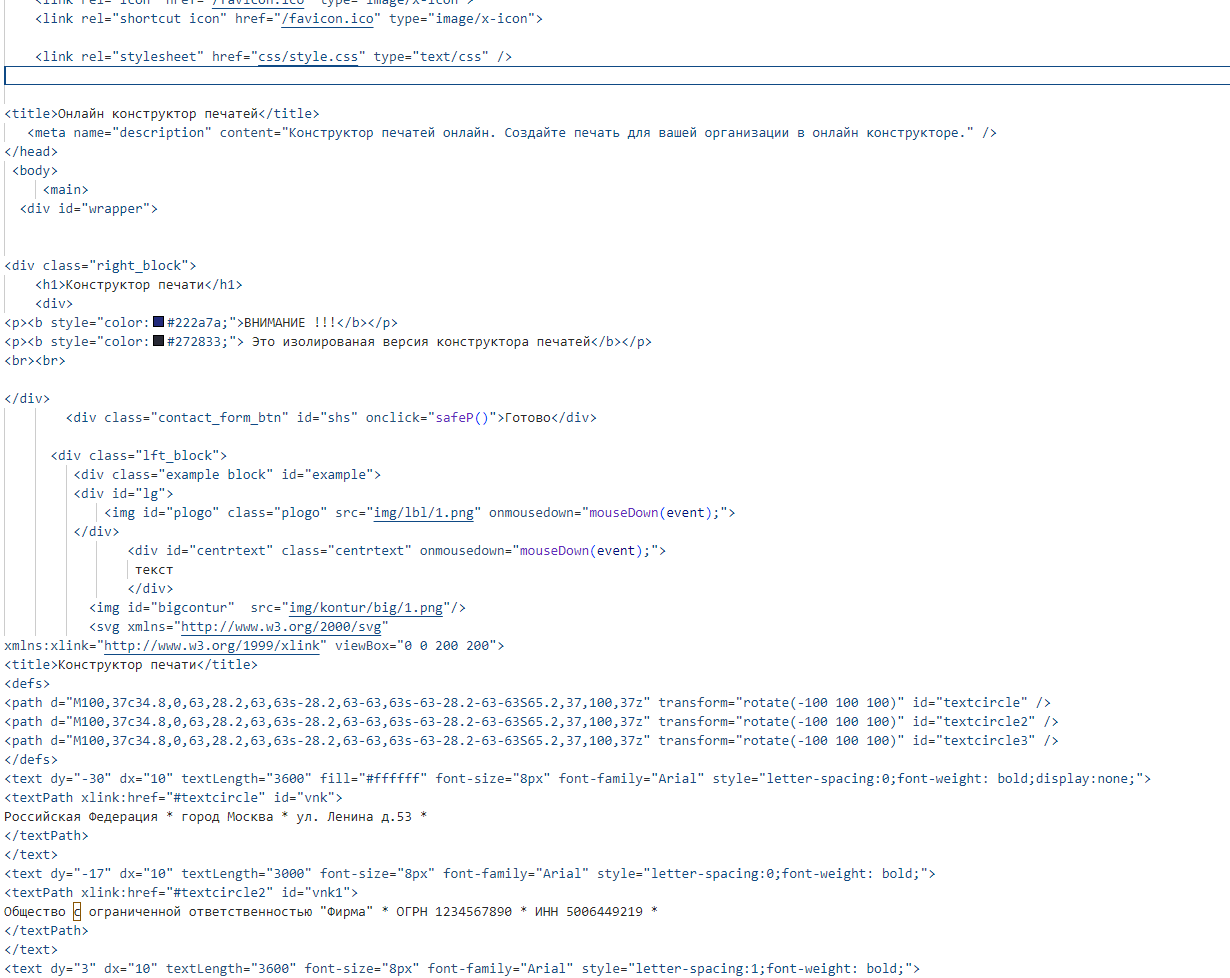


Рисунок 2 – Верстка, обозначение положения для различных элементов печати.

*Приложение 2*

Рисунок 3 — Код HTML верстки главной страницы



Рисунок 4 — Код HTML верстки главной страницы