**Введение**

Данный отчет описывает разработку интернет-ресурса для приема заказов на услуги по перетяжке мягкой мебели. Проект обеспечивает создание удобного и функционального веб-сайта, позволяющего клиентам легко оформлять заказы и узнавать информацию об услугах. Реализованные в рамках проекта функции обеспечивают эффективное взаимодействие с клиентами и оптимизацию процессов обработки заказов.

Описание включает этапы проектирования интерфейса, разработки функционала, тестирования и внедрения. Особое внимание уделяется удобству использования сайта для пользователей и интеграции с системами управления логистикой.

Отчет содержит подробные описания всех этапов разработки:

1 Постановка задачи

В этом разделе описывается организационно-экономическая сущность проекта. Указывается наименование задачи, цель разработки и назначение программного продукта (ПП). Также рассматривается периодичность использования сайта и его связь с другими задачами.

2 Функциональные требования

Перечисляются основные функции, которые должен выполнять будущий ПП. Указываются возможности для двух типов пользователей: гостей и администраторов. Гости могут просматривать сайт и заполнять заявки, а администраторы – обновлять информацию и взаимодействовать с пользователями.

3 Описание входной,выходной и условно-постоянной информации .

4 Эксплуатационные требования

Описываются требования к производительности, надежности и интерфейсу. Определяются критерии времени загрузки, количество пользователей и дизайн интерфейса.

5 Диаграмма вариантов использования

Представляется диаграмма, отражающая взаимодействие пользователей с системой в соответствии с функциональными требованиями.

6 Выбор стратегии разработки и модели жизненного цикла (ЖЦ)

Проводится сравнительный анализ различных моделей ЖЦ. На основе критериев выбирается наиболее подходящая модель для реализации проекта.

9 Инструменты разработки

Приводится список используемых инструментов и технологий. Описываются возможности платформы WordPress, плагинов WPForms и Elementor, а также других средств разработки.

10 Технические характеристики

Указываются параметры оборудования и программного обеспечения, на котором будет реализован проект.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки интернет-ресурса. Здесь можно узнать об организации данных в контексте среды разработки. В данном разделе будет описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации.

«Реализация» – это третий раздел отчета, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения.

Четвёртый раздел – «Тестирование». В нем будет описано полное и функциональное тестирование данной программы. Будут смоделированы все возможные действия пользователя при работе с web-ресурсом, начиная от входа на сайт заканчивая закрытием вкладки.

В разделе «Руководство пользователя» будет описано назначение, область применения, среда функционирования данного программного продукта.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств.

В разделе «Список использованных источников» будет приведён список используемых при разработке источников.

В приложении А будет представлена диаграмма вариантов использования, диаграмма последовательности, диаграмма деятельности, структура сайта и диаграмма компонентов.

В приложении Б будет представлены UX UI прототипы.

В приложении В будет представлены тест- кейсы.

1. **Постановка задачи**
   1. **Организационно-экономическая сущность задачи**

Наименование задачи:Разработка интернет ресурса для заказа услуг по

ремонту и перетяжке мягкой мебели

Цель разработки:Создание сайта для ознакомления с компанией и заказа услуг по ремонту и перетяжке мягкой мебели

Назначение: ПП разрабатывается с целью создания удобной и надежной онлайн платформы, которая предоставляет возможность заказать услугу по перетяжке мягкой мебели.

Периодичность использования: Пользователи могут посещать сайт и осуществлять торговые операции в любое удобное для них время в зависимости от своих потребностей. Сайт доступен для использования круглосуточно, 7 дней в неделю. Периодичность использования сайта зависит от активности пользователей, в зависимости от потребностей и предпочтений каждого отдельного пользователя.

* 1. **Функциональные требования**

Описание перечня функций и задач, которые должен выполнять будущий ПП:

Гость:

* просмотр страниц сайта;
* просмотр социальных сетей;
* ознакомление с услугами по ремонту и перетяжке мягкой мебели;
* заполнение заявки по заказу ремонту и перетяжке мягкой мебели;

Администратор:

* обновление информации.
  1. **Описание входной информации**
* email;
* название фирмы;
* требования к перевозке.
  1. **Описание выходной информации**
* подтверждение отправки формы обратной связи;
* письмо на email с ответом.
  1. **Описание условно-постоянной информации**
* контактная информация;
* фотографии;
* название компании.
  1. **Эксплуатационные требования**

требования к применению:помогает быстро ознакомиться с компанией и заказать услугу по ремонту и перетяжке мягкой мебели;

требования к производительности: для хорошей производительности время загрузки страницы на скорости интернета 3G не должно превышать 3 секунд. Одновременное обслуживание не менее 500 активных пользователей без потери производительности, а так же экономичное потребление серверных ресурсов, включая оперативную память и процессорное время.

требования к реализации:для реализации статических страниц и шаблонов должны использоваться языки HTML и CSS. Для реализации интерактивных элементов клиентской части должны использоваться языки JavaScript. Для реализации динамических страниц должен использоваться язык PHP.

требования к надежности:система может быть недоступна не более чем 24 часа в год. У администратора приложения должна быть возможность выгрузить и загрузить копию**.**

требования к интерфейсу:при разработке сайта должен быть лаконичный и подробный дизайн, который будет привлекательным для пользователей, с удобной навигацией преимущественного светлых цветов. Основные разделы сайта должны быть доступны с первой страницы. Удобная навигация играет ключевую роль в опыте пользователя, поэтому основные разделы сайта должны быть легко доступны с первой страницы. Это поможет пользователям быстро ориентироваться на сайте и находить необходимую информацию без лишних усилий. Грамотный пользовательский интерфейс включает в себя интуитивно понятные элементы управления, понятную структуру страниц и легкость взаимодействия с контентом. Важно также обеспечить адаптивность сайта под различные устройства, такие как компьютеры, телефоны и планшеты, чтобы пользователи могли комфортно пользоваться сайтом независимо от устройства, которое они используют.

* 1. **Диаграмма вариантов использования**

В диаграмме представлены возможности гостя и администратора, которые соответствуют функциональным требованиям (пункту 1.2)

Диаграмма представлена в приложении А

* 1. **Выбор стратегии разработки и модели ЖЦ**

Таблица 1 **–** Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерий требования | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да | Нет | Нет | Нет |
| 2. | Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| 3. | Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Да |
| 4. | Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |

Продолжение таблицы 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |
| 8. | Итого: | 3 | 3 | 4 | 4 | 4 | 4 |

Итог: на основе результатов заполнения табл. 3 подходящей является RAD, инкрементная, быстрого прототипирования и эволюционная модель.

Таблица 2 **–** Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерий категории команды разработчиков проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | Да | Да |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | Да |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | Нет | Да | Нет | Нет |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Нет | Да | Да | Да |

Продолжение таблицы 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 7. | Итого: | 4 | 4 | 3 | 4 | 3 | 2 |

Итог: На основе результатов заполнения табл. 4 подходящими являются каскадная, v-образная, и инкрементная модель.

Таблица 3 **–** Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерий категории команды разработчиков проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да | Нет | Да | Нет | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет | Да | Нет | Да | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | Да | Да |
| 5. | Итого: | 1 | 1 | 2 | 2 | 3 | 1 |

Итог: На основе результатов заполнения табл. 5 подходящей является модель быстрого прототипирования

Таблица 4 **–** Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерий категории команды разработчиков проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Разрабатывается ли в проекте продукт нового для организации направления? | Да | Да | Нет | Да | Нет | Да |

Продолжение таблицы 4

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 2. | Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | Нет | Нет |
| 3. | Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Ожидается ли длительная эксплуатация продукта? | Да | Да | Нет | Да | Да | Да |
| 5. | Необходим ли высокий уровень надежности продукта проекта? | Нет | Да | Нет | Да | Нет | Да |
| 6. | Предполагается ли эволюция продукта проекта в течение ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет | Да | Да | Да |

Итог: На основе результатов заполнения табл. 6 подходящими являются v-образная, инкрементная и эволюционная модель.

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является инкрементная.

13 каскадная модель

14 V-образная модель

13 RAD модель

16 инкрементная модель

14 быстрого прототипирования модель

13 эволюционная модель.

* 1. **Инструменты разработки**

Для создания сайта используется платформа WordPress. Благодаря большому выбору тем и плагинов можно быстро создать функциональный и красивый сайт. Для создания сайта на WordPress используется его движок, темы оформления и плагины. Движок — это основа сайта, обеспечивающая работу с контентом, настройками и функционалом. Он основан на языке программирования **PHP** и использует **базу данных MySQL** для хранения информации (посты, страницы, пользователи и настройки). В основные функции движка входит: управление контентом (поддержка различных типов контента: текст, изображения, видео, аудио, галереи; категоризация контента через рубрики и теги), пользовательские роли и доступ (разделение прав доступа через систему ролей администратор, редактор, автор, подписчик; возможность настроить доступ к различным разделам сайта для пользователей), гибкость настройки(Поддержка тем для изменения дизайна сайта; возможность расширения функционала через плагины), мультиязычность(поддержка создания сайтов на нескольких языках; возможность установки плагинов для автоматического или ручного перевода). Особенности движка: открытый исходный код (движок WordPress распространяется под лицензией GPL, что позволяет свободно изменять и адаптировать его под свои нужды), широкая экосистема (мощное сообщество разработчиков активно поддерживает платформу и создает новые решения), регулярные обновления (WordPress постоянно совершенствуется, что обеспечивает высокую безопасность и актуальность технологий), масштабируемость (движок подходит как для небольших сайтов, так и для крупных проектов с высокой посещаемостью). Тема задает дизайн, расположение элементов интерфейса и базовые стили сайта. Плагины расширяют функционал сайта. WordPress. В проекте использовались такие плагины как WPForms и Elementor. Плагин Elementor позволяет внести внешние изменения в дизайн сайта, добавить на страницу фото, текст, ссылки, кнопки, видео и т.д. Плагин WPForms отвечает за создание форм на сайте, при чем самым удобным и простым способом. Так же используются иные инструменты:

* WEB-ресурс DRAW.IO, который служит для создания графической части и разработки UML-диаграмм;
* Microsoft Office Word – для написания документации к программному продукту;
* PhpMyAdmin 6.0.0 – будет использоваться в качестве инструмента для администрирования СУБД MySQL;

Разработка проекта будет происходить на компьютере со следующими параметрами:

* процессор 12th Gen Intel(R) Core(TM) i5-12450H 2.00 GHz;
* объем оперативной памяти 16.00 GB;
* объем места на жестком диске 500 GB;
* ОС Windows 11
  1. **Составление** **плана и графика работы над проектом**

Составление плана и графика работы над проектом представлено в диаграмме Ганта, она визуально отображает план проекта во времени, показывая задачи и их продолжительность, позволяя отслеживать прогресс и управлять сроками

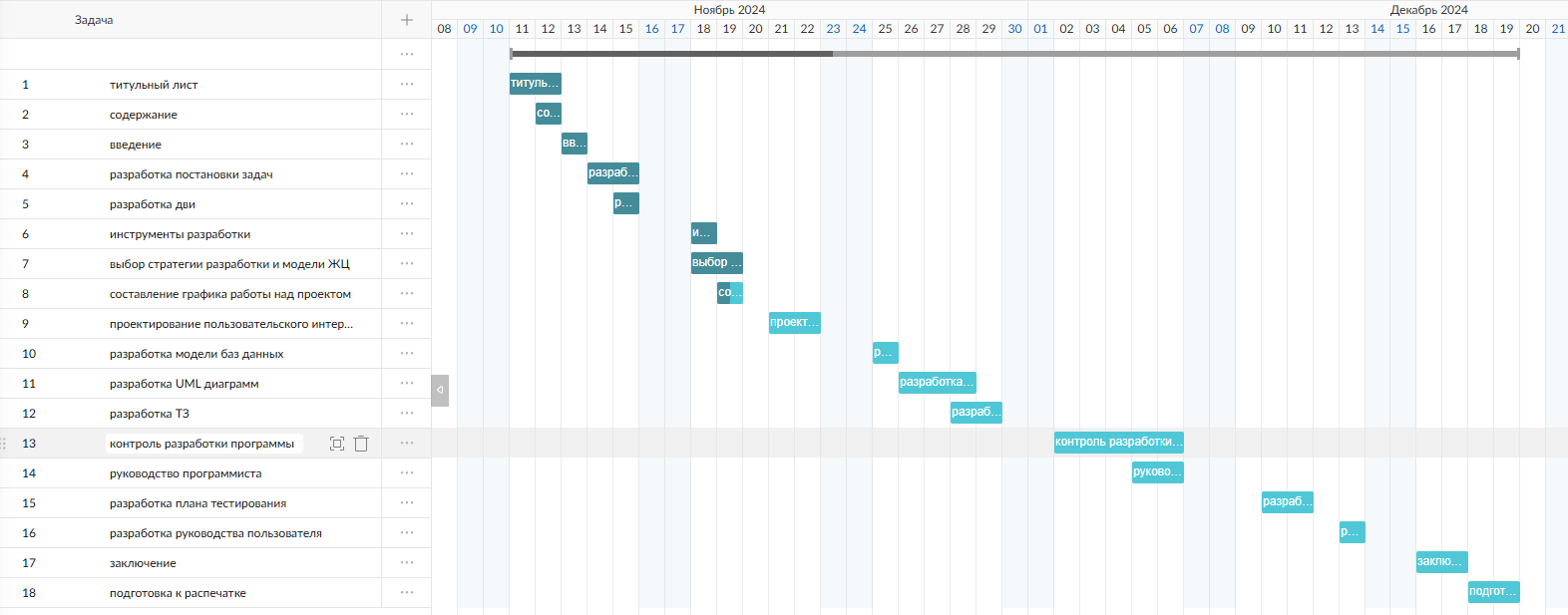


Рисунок 1 – Диаграмма Ганта

1. **Проектирование задачи**
   1. **Разработка структуры сайта**

Структура сайта представляет из себя навигацию сайта, которая предоставлена в приложении А.

В меню сайта представлено 3 страниц :

* главная;
* о нас;
* связь с нами;

С главной страницы можно перейти на страницы:

* о нас;
* связь с нами.
  1. **Разработка UML-диаграмм**
     1. **Диаграмма деятельности**

В диаграмме деятельности представлены действия, которые можно выполнить на каждой из страниц.

Разработанная диаграмма представлена в приложение А.

* + 1. **Диаграмма компонентов**

Диаграмма компонентов – это UML статическая структурная диаграмма, которая показывает разбиение программной системы на структурные компоненты и связи между компонентами. Данную диаграмму рассмотрим в приложении А.

* + 1. **Диаграмма последовательности**

Диаграмма последовательности – UML диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта и взаимодействие актеров информационной системы в рамках прецедента. Диаграмму последовательности можно рассмотреть в приложении А.

* 1. **Разработка пользовательского интерфейса**

**Разработка пользовательского интерфейса (UI) для сайта** — это процесс создания визуальной части сайта, с которой взаимодействует пользователь. Основная цель UI — сделать сайт интуитивно понятным, привлекательным и удобным для использования.

Этот процесс начинается с анализа целевой аудитории, включая изучение предпочтений, целей и привычек пользователей, а также анализа конкурентов и актуальных трендов в дизайне. Далее создается структура сайта, разрабатываются карты сайта и пользовательские потоки. После этого начинается этап визуального дизайна, включающий выбор цветовой палитры, шрифтов, стиля кнопок, иконок и других элементов интерфейса. Визуальный прототип должен включать все элементы интерфейса и демонстрировать их взаимодействие. Интерактивные элементы, такие как всплывающие окна, анимации и эффекты при наведении, делают сайт более динамичным. После завершения дизайна проводится тестирование, включая проверку удобства использования, обратную связь от пользователей и тесты на различных устройствах и браузерах. Завершающий этап — это передача готовых дизайнов разработчикам и контроль за реализацией, чтобы сайт соответствовал изначальным макетам. В процессе разработки важно придерживаться принципов простоты, последовательности, интуитивности, доступности и скорости загрузки. Эти аспекты делают интерфейс удобным, понятным и привлекательным для пользователей, что положительно влияет на их опыт взаимодействия с сайтом.

Прототип – это наглядная модель пользовательского интерфейса. В сущности, это «черновик», созданный на основе представления разработчика о потребностях пользователя. Итоговое отображение программы может отличаться

от прототипа. Прототипы UX представлены в приложении И. Прототипы UI представлены в приложении Б.

1. **Реализация**
   1. **Руководство программиста**

Программный продукт разработан с использованием WordPress. Этот ресурс представляет собой удобный и простой в использовании инструмент, который предлагает готовые темы. Для начала работы в конструкторе необходимо создать или войти в свою учетную запись. После этого можно выбрать готовую тему. Создание страниц осуществляется при помощи кнопки «Создать страницу», которая позволяет включить страницу в меню сайта. Редактировать страницы можно благодаря плагину Elementor.

* + 1. **Создание сайта**

Для создание сайта на WordPress необходимо скачать его и зарегистрироваться.

Данные для входа на сайт администратора:

* логин: admin;
* пароль: admin1234567\_56fhskg.
  + 1. **Настройка функциональности и дизайна**

Настройка функциональности и дизайна WordPress — это один из ключевых этапов разработки сайта, от которого зависит удобство работы с ресурсом и его визуальная привлекательность. Для расширения функциональности сайта используются плагины, такие как “Elementor” для редактирования дизайна и WPWorms для создания форм обратной свзяи. После базовых настроек функциональности переходим к дизайну сайта. Важным шагом является выбор подходящей темы. В разделе "Внешний вид" можно установить темы, адаптированные для мобильных устройств и поддерживающие современные стандарты. WordPress дает множество тем для создания сайта, которые могут быть отредактированы (изменен фон, поля ввода и т.д.), дополнены новыми элементами и удалены.

* + 1. **Редактирование страниц**

Для создания страницы необходимо в меню найти кнопку «Страницы», далее «Добавить страницу».

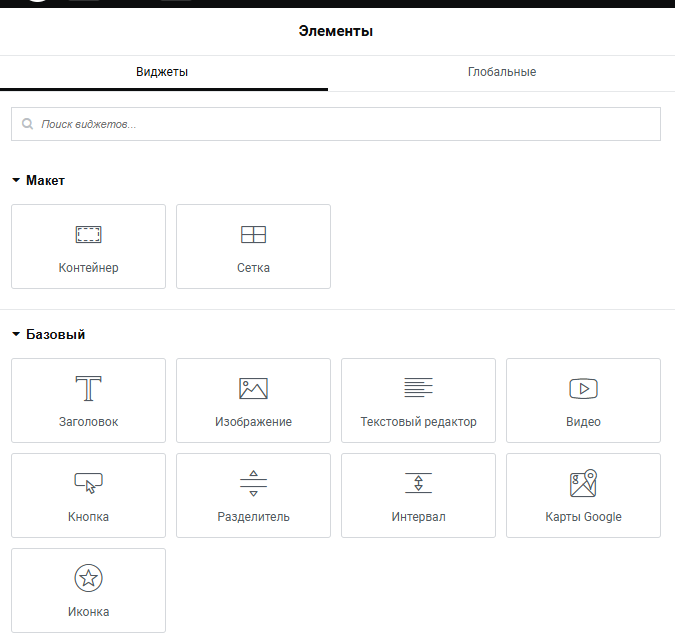
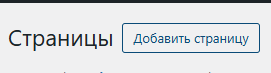
****После создания страницы можно приступать к её дизайну. Для этого используется плагин “Elementor”. Добавляйте **разделы** (секции), **колонки** и **виджеты**: текстовые блоки, заголовки, кнопки, изображения и т.д. Перетащите нужные виджеты (например, "Картинка", "Текстовый редактор", "Кнопка") в рабочую область. Настройте стиль каждого элемента: измените цвета, шрифты, размеры и эффекты анимации.

Рисунок 3 – Панель виджетов

Рисунок 2 - Создание страницы

Для создания меню нужно перейти в “Внешний вид” , затем в “меню”. Добавляем созданные страницы и настраиваем их порядок.

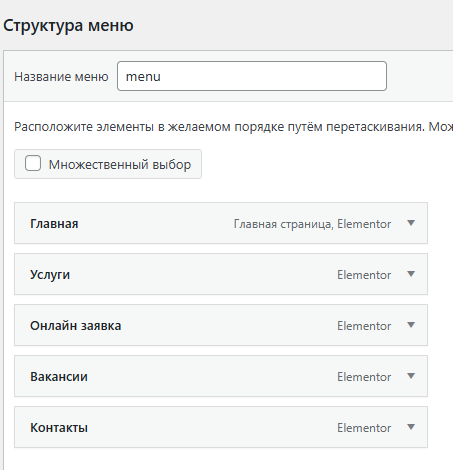
* + 1.  **Редактирование главной страницы сайта**

Рисунок 4 – Структура меню

Чтобы редактировать главную страницу сайта, необходимо найти кнопку

“Страницы” – “все страницы”.

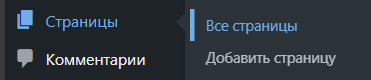
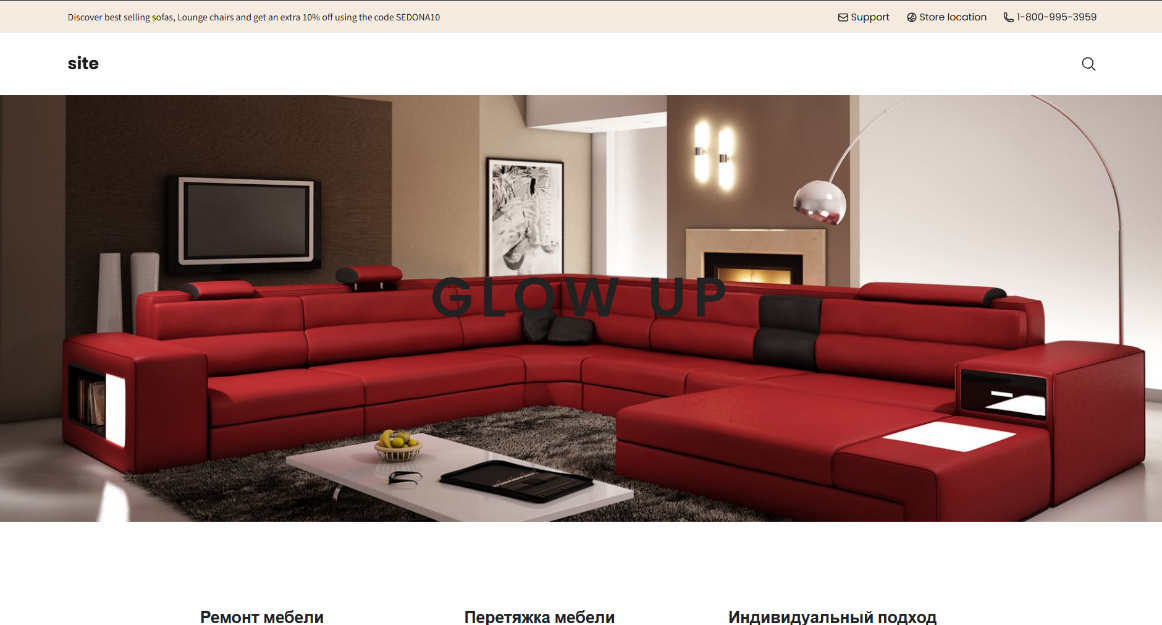


Рисунок 5 – Все страницы

Для редактирования, необходимо выбрать любую страницу и нажать редактировать в Elementor.

Рисунок 6 – главная страница



1. **Тестирование** 
   1. **Тесты на использование**

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения испытания реализации программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Разработанные тест-кейсы и статус их выполнения представлены в приложение В.

Таблица 5 - Расписание работ над проектом

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Дата | Деятельность | Продолжительность, ч |
| Левицкий Максим | 12.12.2024 | Разработка тестов | 2 ч |
| Левицкий Максим | 13.12.2024 | Тестирование сайта | 1 ч |
| Левицкий Максим | 14.12.2024 | Составление о найденных дефектах | 1 ч |
| Левицкий Максим | 16.12.2024 | Исправление найденных ошибок | 3 ч |
| Левицкий Максим | 17.12.2024 | Проведение регрессионного тестирования | 1 ч |
| Левицкий Максим | 18.12.2024 | Составление отчета о результатах тестирования | 1 ч |

**4.2 Отчёт о результатах тестирования**

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

Таблица 6 – Исправность ПП

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | Важность | | | |
| Статус | Количество | Низкая | Средняя | Высокая | Критическая |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Найдено | 0 | 0 | 0 | 0 | 0 |
| Исправлено | 0 | 0 | 0 | 0 | 0 |

Продолжение таблицы 6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 |
| Проверено | 0 | 0 | 0 | 0 | 0 |
| Открыто заново | 0 | 0 | 0 | 0 | 0 |
| Отклонено | 0 | 0 | 0 | 0 | 0 |

1. **Руководство пользователя**

Для того, чтобы открыть сайт необходимо перейти по ссылке.

После открытия сайта загружается главная страница, представленная на рисунке 7.

Рисунок 7 – Главная страница



Переход по страницам можно осуществлять по горизонтальному меню. Которое представлено на рисунке 8.

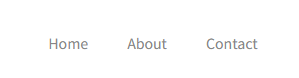


Рисунок 8 – меню сайта

Главная страница включает в себя информацию “о нас” и перечень услуг, предлагаемые компанией. Данная страница предоставлена на рисунке 9.

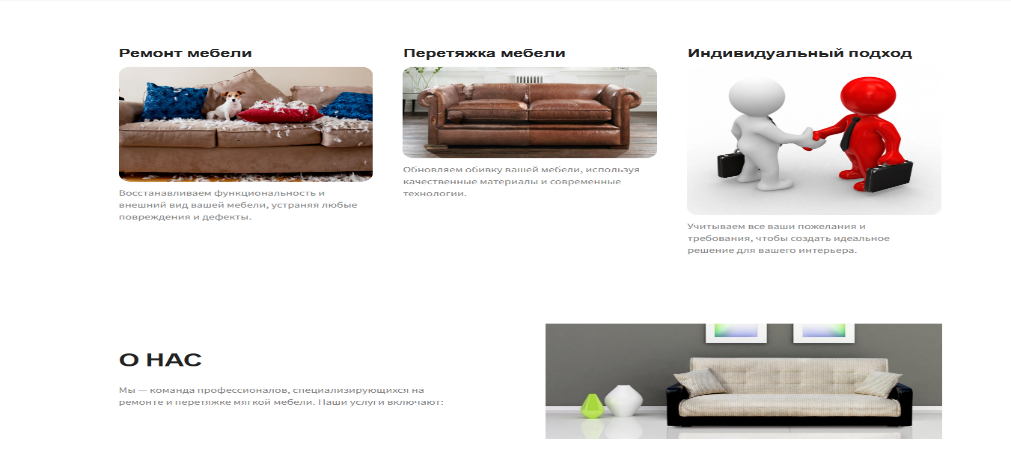


Рисунок 9 – Главная страница

Также можно связаться с нами перейдя на страницу “Контакты”, представленную на рисунке 10.

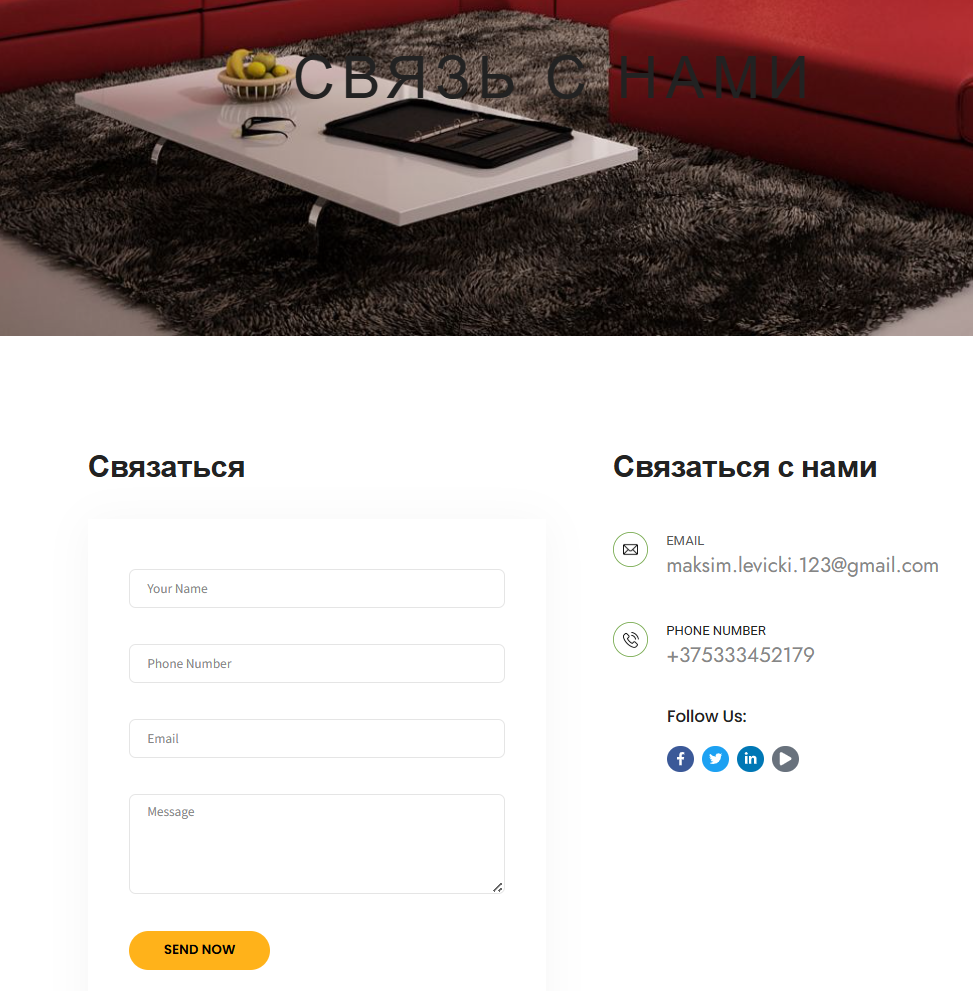
****

Рисунок 10 – Страница Контакты

**Заключение**

Целью данного проекта является разработать интернет-ресурс для приема заказов на услуги грузоперевозок. Обеспечение удобного и функционального веб-сайта, позволяющего клиентам легко оформлять заказы и узнавать информацию об услугах.

В ходе реализации поставленной задачи были закреплены знания по использованию конструктора WordPress, а также курс предмета «Технология разработки программного обеспечения».

Поставленная задача выполнена в соответствии с требованиями заказчика, созданы и протестированы все необходимые страницы и компоненты проекта.

В ходе тестирования все исключительные ситуации были обработаны. Проект работает без сбоев и ошибок. В поставленной задаче был реализован

простой и понятный пользовательский интерфейс.

Исходя из этого, можно сделать вывод, что программа реализована успешно.

**Список использованных источников**

1 [Электронный ресурс]. Режим доступа - <https://www.phpmyadmin.net/?authuser=0>

2 Как создать сайт на WordPress [Электронный ресурс]. Режим доступа -<https://youtu.be/f6cimPl2OSY?si=YmLL6JgnGW5SadbU> .

3Создание сайта [Электронный ресурс]. Режим доступа - [(905) Создание сайта на WordPress за 5 минут (пошаговая установка wordpress на хостинге) - YouTube](https://www.youtube.com/watch?v=63auV6kgXDM)