**Введение**

Цель данного проекта — создание функционального и визуально привлекательного веб-сайта с удобным и интуитивно понятным интерфейсом, разработанного на основе предоставленного макета. В процессе реализации планируется использовать современные веб-технологии, такие как HTML, CSS и JavaScript, для разработки клиентской части (frontend) сайта. Основная задача проекта — создание пользовательского интерфейса для сайта, предоставляющего услуги по ремонту квартир.

Веб-сайт по ремонту квартир позволяет пользователям выбрать и заказать необходимые услуги от различных компаний и специалистов, не покидая дома. Также предусмотрена возможность получить профессиональную консультацию по всем вопросам, оформить заказ на ремонт по индивидуальному проекту, рассчитать стоимость работ и ознакомиться с примерами ранее выполненных проектов.

Главная страница сайта должна быть адаптирована для корректного отображения на устройствах с различными разрешениями экрана, обеспечивая комфортный просмотр как на стационарных компьютерах, так и на мобильных устройствах.

Ниже приведено краткое описание основных разделов пояснительной записки:

Первый раздел — «Анализ задачи». В нём представлена постановка задачи и описание используемых инструментов разработки. Также рассматривается, как подобные задачи решаются в настоящее время.

Раздел «Проектирование задачи» освещает ключевые аспекты разработки программного продукта. В нём приведены рекомендации по стилю и тест-кейсы.

Раздел «Реализация» описывает процесс вёрстки, создание динамических элементов с помощью CSS и JavaScript, а также структуру и спецификацию файлов проекта. В этом разделе также содержатся такие подразделы, как руководство программиста и спецификация файлов.

Четвёртый раздел — «Тестирование» — содержит отчёт о тестировании программного продукта. В рамках тестирования моделируются возможные сценарии взаимодействия клиента с приложением.

В разделе «Применение» будет приведена ссылка на размещённый проект.

«Заключение» включает краткое описание поставленной задачи, итоги выполненной работы, описание использованных методов и технологий, а также оценку уровня автоматизации процессов на различных этапах разработки.

В разделе «Список использованных источников» представлен перечень источников, использованных в процессе разработки.

В Приложении А приведены модульные сетки для desktop- и мобильной версий главной страницы сайта.

**1. Анализ задачи**

* 1. **Постановка задачи**

В рамках учебной практики по WEB-программированию передо мной была поставлена задача: сверстать шаблон главной страницы сайта на тему «Ремонт квартир», используя макет, представленный в Figma.  
Ссылка на макет:

https://www.figma.com/design/k8GPXLtLMn2KSxkvnqQQir/Многостраничный-сайт---Ремонт-квартир?node-id=0-1&p=f&t=OjnIseB2MzLQTm7y-0

Основная цель проекта — разработать адаптивную веб-страницу с использованием HTML, CSS и JavaScript, которая будет отображать информацию о предоставляемых услугах по ремонту квартир.

Верстка должна быть адаптирована для корректной работы на устройствах с различными разрешениями экрана, включая:

* стационарные компьютеры и ноутбуки;
* планшеты;
* мобильные телефоны.

Одним из обязательных этапов является подготовка гайда по стилю — документа, включающего основные правила визуального оформления сайта.  
Гайд служит основой для поддержания единого дизайна на всём сайте и должен содержать следующие элементы:

* логотип;
* цветовую схему;
* используемые шрифты (типографику);
* сетку и отступы;
* структуру меню и навигации;
* оформление кнопок и полей ввода;
* визуальные компоненты;
* пользовательские формы;
* стили для модальных окон.

После завершения базовой вёрстки необходимо приступить к стилизации элементов с помощью CSS, применяя следующие визуальные эффекты:

* эффекты при наведении (hover);
* стилизация ссылок;
* добавление теней;
* использование градиентов;
* оформление изображений;
* оформление пользовательских форм;
* применение анимаций и плавных переходов;
* декоративные элементы и прочие стилистические решения.

Завершающим этапом является добавление динамических и интерактивных элементов с использованием JavaScript и библиотек jQuery и jQuery UI.  
Функциональность должна включать:

* работу с всплывающими окнами;
* создание выпадающего меню и навигационных элементов;
* реализацию слайдеров;
* обработку изображений средствами JS;
* использование возможностей jQuery и jQuery UI;
* применение дополнительных инструментов и методов для реализации интерактивного поведения элементов.
  1. **Инструменты разработки**

Инструменты разработки представляют собой совокупность программных средств, предназначенных для упрощения процессов создания, отладки и оптимизации веб-сайтов и веб-приложений. Они помогают разработчикам эффективнее анализировать код, устранять ошибки, совершенствовать структуру проекта и улучшать пользовательский опыт. В данном разделе будут рассмотрены основные инструменты, использованные при реализации текущего проекта.

Для вёрстки шаблона макета была выбрана среда разработки Microsoft Visual Studio Code, с установленными следующими расширениями:

* HTML CSS Support — поддержка автодополнения и подсказок для HTML и CSS;
* Russian Language Pack — локализация интерфейса редактора на русский язык;
* Live Server — локальный сервер для быстрой предварительной проверки изменений в браузере в реальном времени;
* Bootstrap 5 & Font Awesome Snippets — удобные сниппеты для ускоренной разработки с использованием фреймворка Bootstrap и библиотеки иконок Font Awesome.

В дополнение к этому, для работы с макетом и оформления интерфейсов использовался инструмент Figma — онлайн-платформа для проектирования интерфейсов, которая предоставляет возможности для прототипирования и совместной работы в режиме реального времени. С помощью Figma были проанализированы структура макета, цветовые схемы, типографика и элементы пользовательского интерфейса.

Для хранения и контроля версий проекта применялся сервис GitHub — одна из крупнейших платформ для размещения исходного кода и совместной работы над проектами. GitHub позволяет отслеживать изменения в коде, работать в команде и управлять историей версий.

Все этапы работы по вёрстке и тестированию шаблона выполнялись на персональном компьютере со следующими техническими характеристиками:

* Процессор: AMD Ryzen 9 7945HX;
* ОЗУ: 16 GB DDR4;
* Память: SSD 1024 GB;
* Операционная система: Windows 11 Home.

Использование современных инструментов разработки в сочетании с мощной аппаратной платформой обеспечило высокую производительность на всех этапах реализации проекта — от анализа макета до финального тестирования и подготовки готового решения.

1. **Проектирование задачи**
   1. **Гайд по стилю**
      1. **Логотип**

Логотип — это графический элемент, служащий для идентификации сайта и повышения его узнаваемости. Он представляет собой стилизованное название проекта или символ, связанный с тематикой сайта.

В рамках данного проекта логотип размещён в верхней части страницы и используется для формирования целостного визуального образа.



Рисунок 1 – Логотип сайта

Модульная сетка для версии сайта на персональном компьютере и мобильных устройствах представлена в Приложении А.

* + 1. **Цветовая палитра**

Цветовая палитра:

* основной цвет: # #1A1A1A (черный);
* вторичный цвет: #333 (темно-серый);
* фоновый цвет: #FFFFFF (светло-серый), #212529(темно-синй);
* дополнительные цвета: # #29abe2 (голубой), #97a1b0(серый ламантин), #000(черный).
  + 1. **Типографика**

Типографика — важнейший инструмент в веб-дизайне, позволяющий гармонично сочетать текст и визуальные элементы для усиления восприятия контента пользователем.

В шаблоне макета «Ремонт квартир» используется следующая типографика:

* Основной шрифт: Inter;
* Размер шрифта: от 12 px до 32 px;
* Цвет текста: #FFFFFF (белый), #1A1A1A (черный), #97a1b0(серый ламантин).
  + 1. **Схема сайта**

Схема сайта представляет собой структуру, состоящую из разделов: «Главная страница», «О компании», «Сертификаты», «Политика компании», «Где работаем», «Гарантии», «Вакансии», «Капитальный ремонт». Сама схема главной страницы сайта представлена в приложении А.

* + 1. **Навигация**

Навигация — одна из ключевых составляющих любого сайта. Простая и понятная структура помогает пользователю быстро находить нужную информацию, повышая удобство использования ресурса. Простое правило: чем проще ориентироваться на сайте, тем быстрее пользователь найдёт необходимую информацию.

В шапке сайта пользователь может перейти к различным разделам сайта, по кнопке «Калькулятор ремонта», а также увидеть контакты и адрес по услугам ремонта. При нажатии на логотип фирмы, можно обновить страницу. Шапку сайта можно рассмотреть на рисунке 2.

Рисунок 2 – Шапка сайта

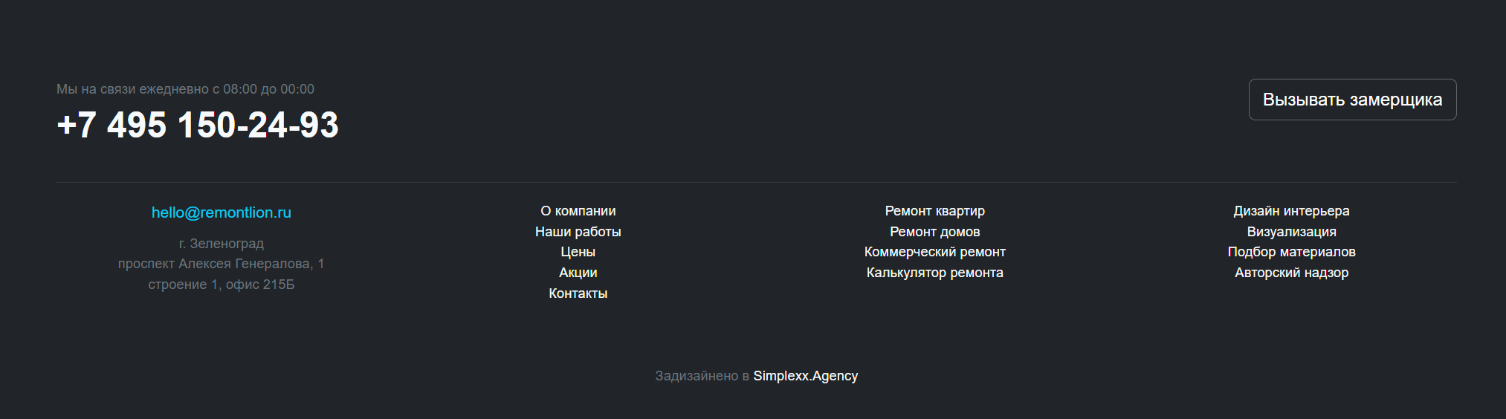
Подвал представлен на рисунке 3.

Рисунок 3 – Подвал сайта

* + 1. **Иконки**

Иконка – это максимально упрощенное изображение объекта или действия.

Иконки идеально вписываются в современные тренды веб-дизайна. Помимо этого, у иконок есть ещё одно неоспоримое преимущество перед - отсутствие языкового барьера. Иллюстрации понимают всё, поэтому иконки используют в операционных системах, в интерфейсах, на сайтах и так далее.

Иконка перехода на страницу инстаграмма компании представлена на рисунке 4.



Рисунок 4 – Иконка перехода на страницу компании

* + 1. **Кнопки и поля ввода**

Интерактивные элементы пользовательского интерфейса создают основу мощных и удобных цифровых продуктов. Внимательный подход к деталям помогает создать положительный опыт взаимодействия с продуктом в целом. Кнопка является одним из самых распространенных элементов любого интерфейса.

Кнопка «Зарегистрироваться» расположена на рисунке 6.

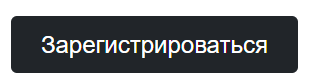


Рисунок 6 – Кнопка «Зарегистрироваться»

Кнопка «Калькулятор ремонта» представлена на рисунке 7.

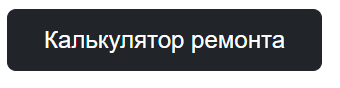


Рисунок 7 – Кнопка «Калькулятор ремонта»

Кнопка «Посмотреть больше проектов» представлена на рисунке 8.

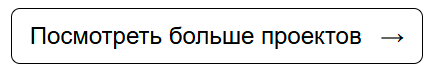


Рисунок 8 – Кнопка «Калькулятор ремонта»

Кнопка «Оставить отзыв» представлена на рисунке 9.

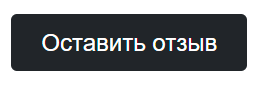


Рисунок 9 – Кнопка «Оставить отзыв»

Кнопка «Вызвать замерщика» представлена на рисунке 10.

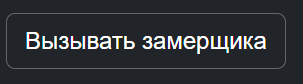


Рисунок 10 – Кнопка «Вызвать замерщика»

Поля ввода, кнопка «Зарегистрироваться» в форме «Регистрация» представлены на рисунке 11.

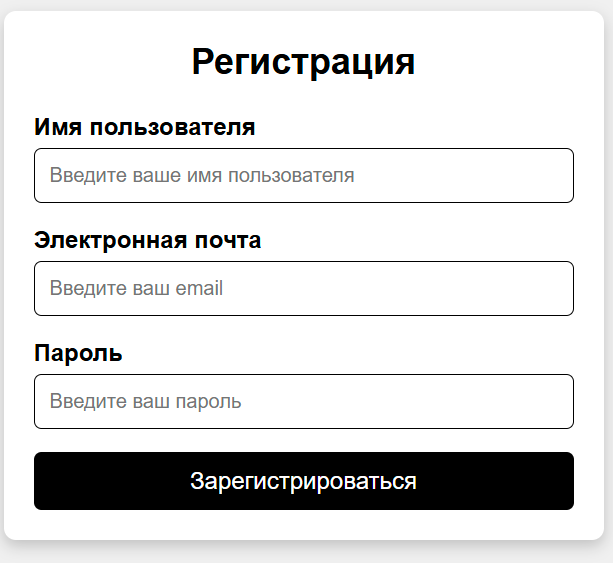


Рисунок 11 – Форма «Регистрация»

Поля ввода, кнопка «Рассчитать» в форме «Калькулятор ремонта» представлены на рисунке 12.

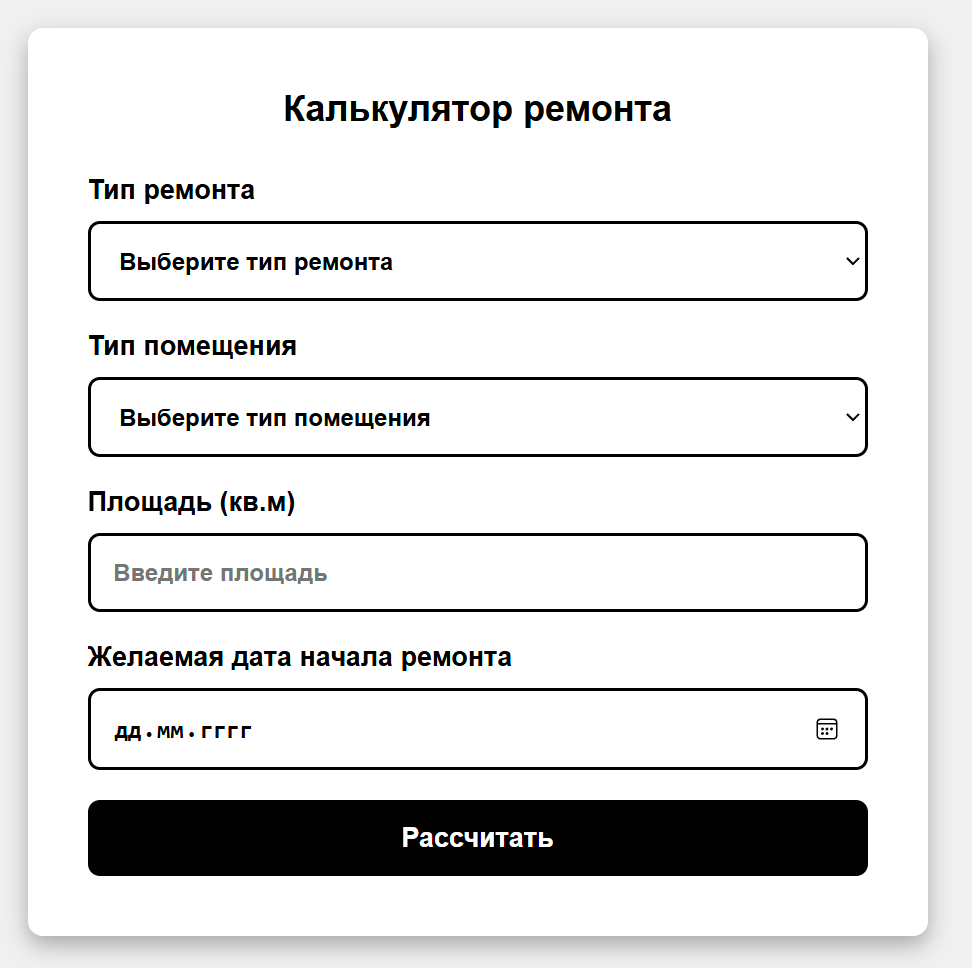


Рисунок 12 – Форма «Калькулятор ремонта»

Поля ввода, кнопка «Отправить» в форме «Оставить отзыв» представлены на рисунке 13.

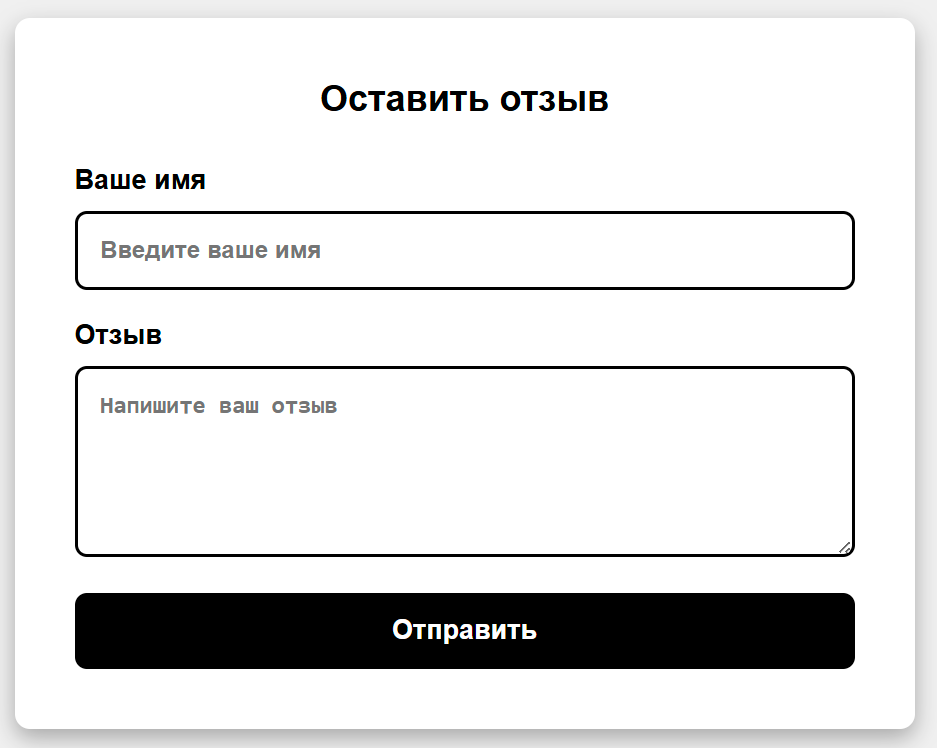


Рисунок 13 – Форма «Оставить отзыв»

Поля ввода, кнопка «Отправить» в форме «Вызвать замерщика» представлены на рисунке 14.

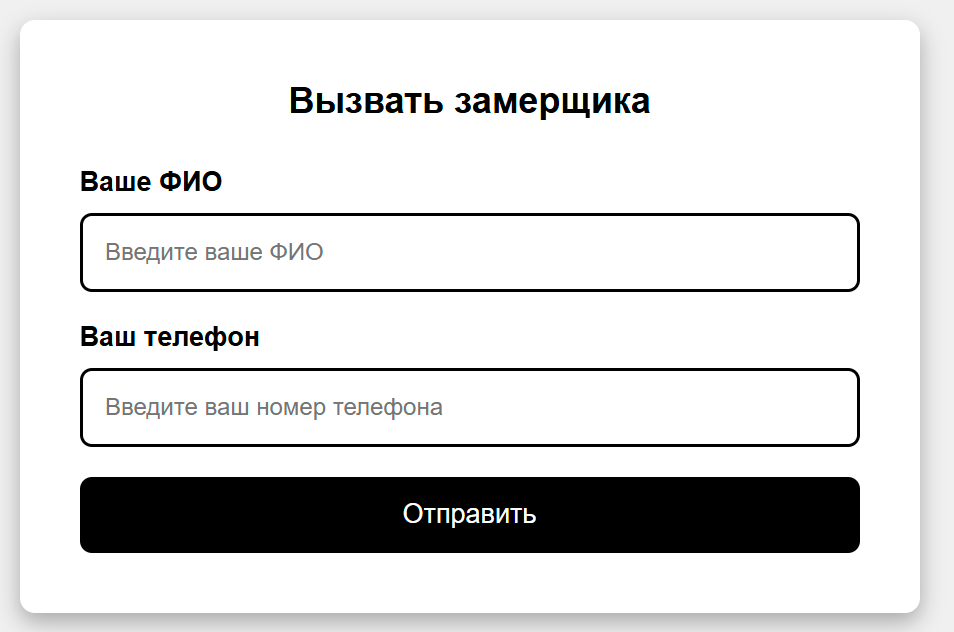


Рисунок 14 – Форма «Вызвать замерщика»

* + 1. **Компоненты**

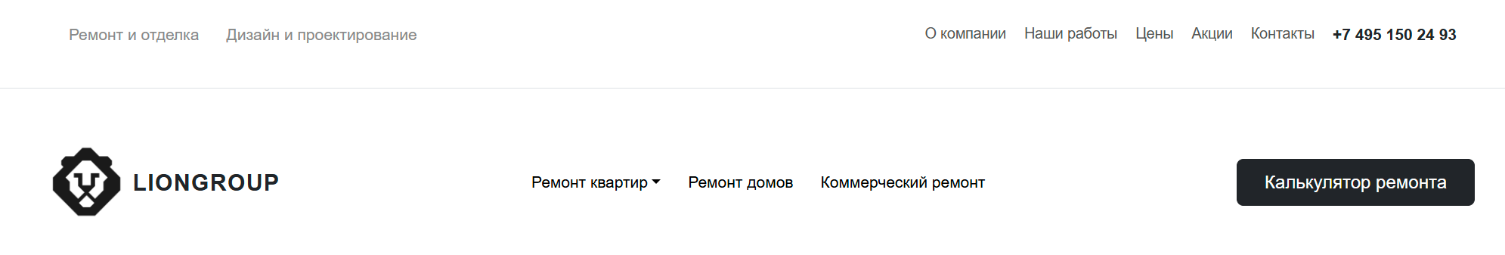
В разрабатываемом интернет-ресурсе можно выделить блоки, пользовательские формы и модальные окна представленные на рисунках 15-25.

Рисунок 15 – Шапка сайта

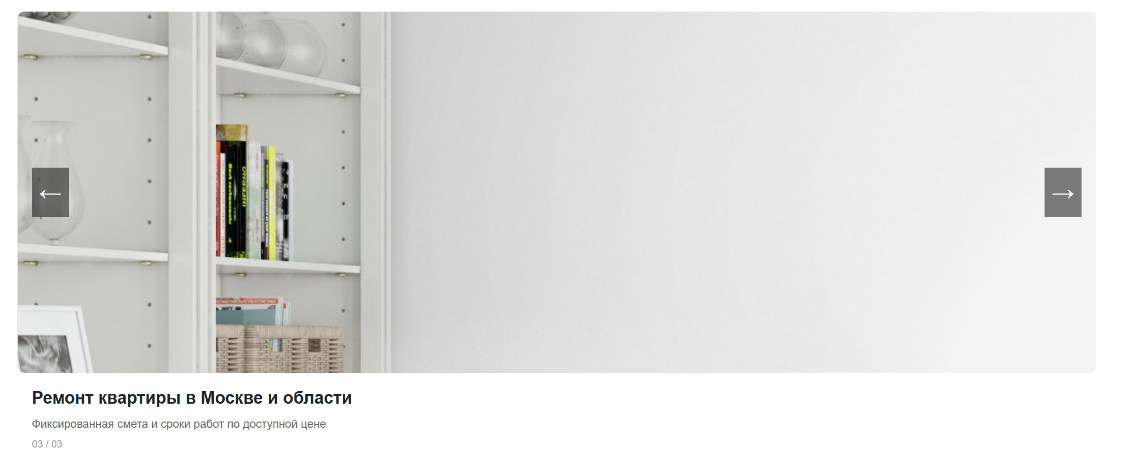


Рисунок 16 – Блок «Слайдер»

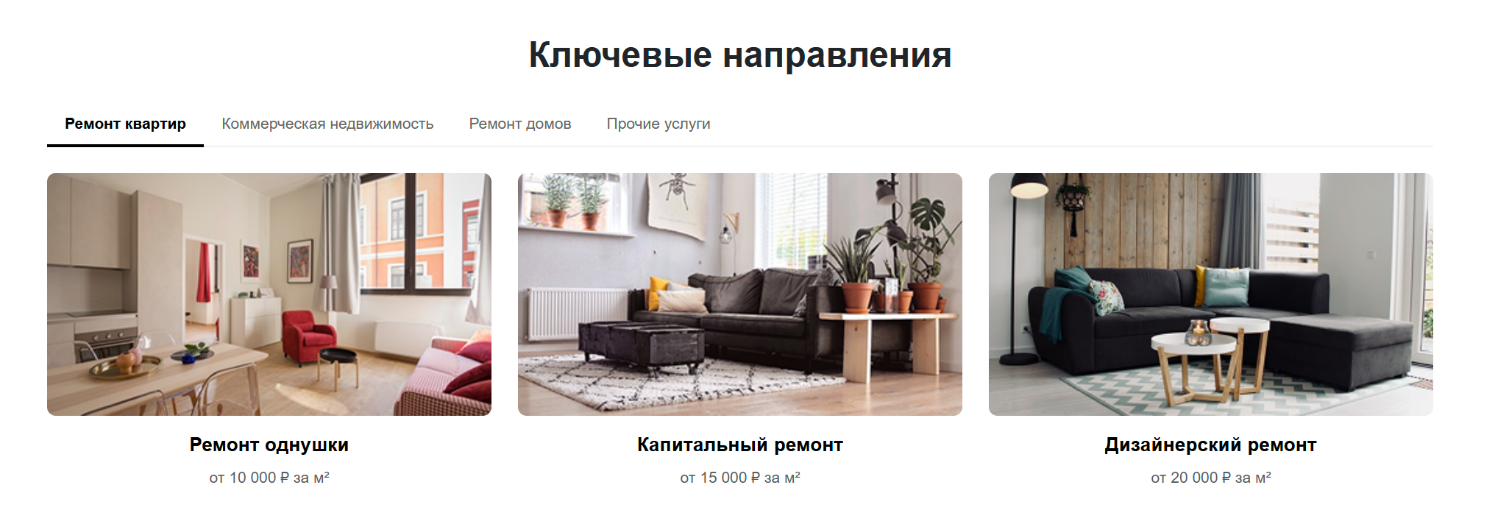


Рисунок 17 – Блок «Ключевые направления»

Рисунок 18 – Блок «При работе с нами вы получаете»

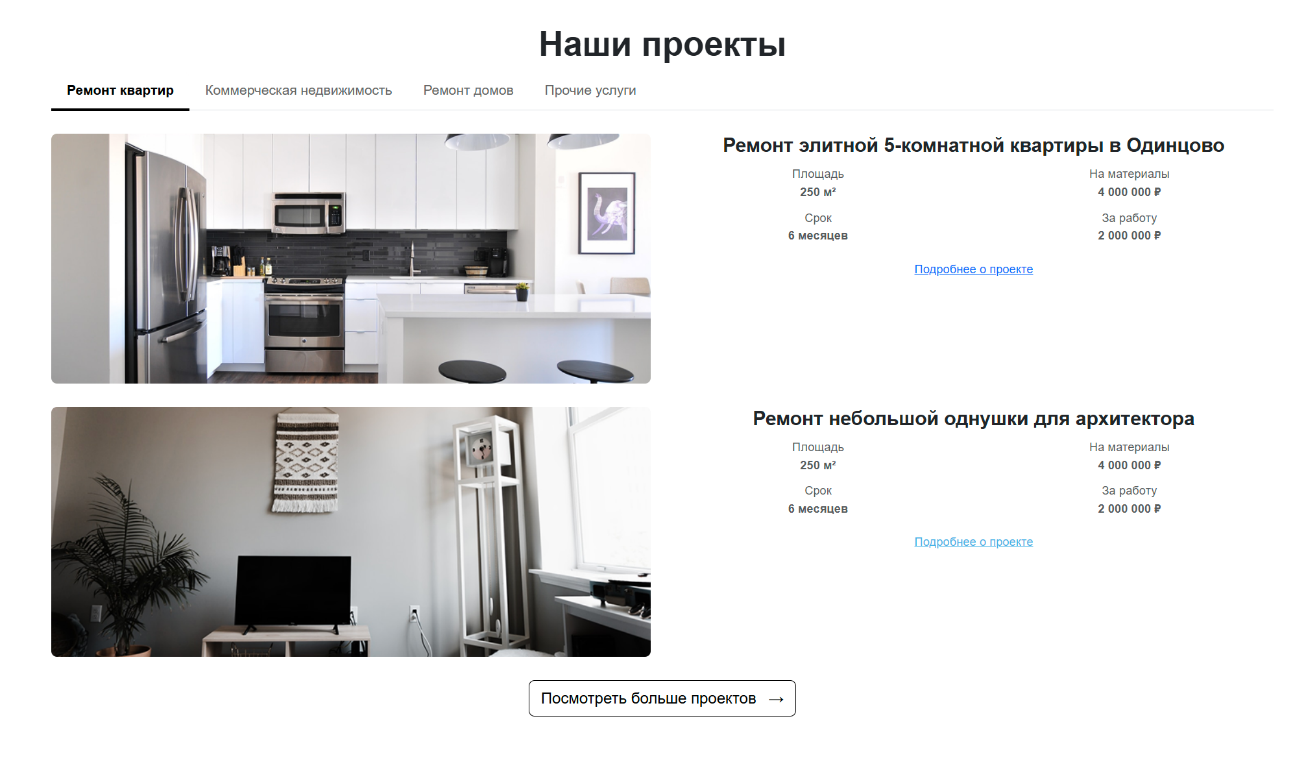


Рисунок 19 – Пользовательская форма «Наши проекты»

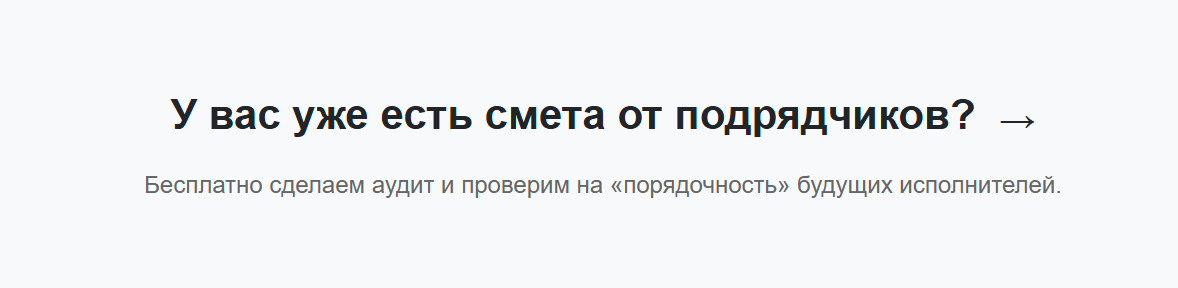


Рисунок 20 – Блок «У вас уже есть смета от подрядчиков»

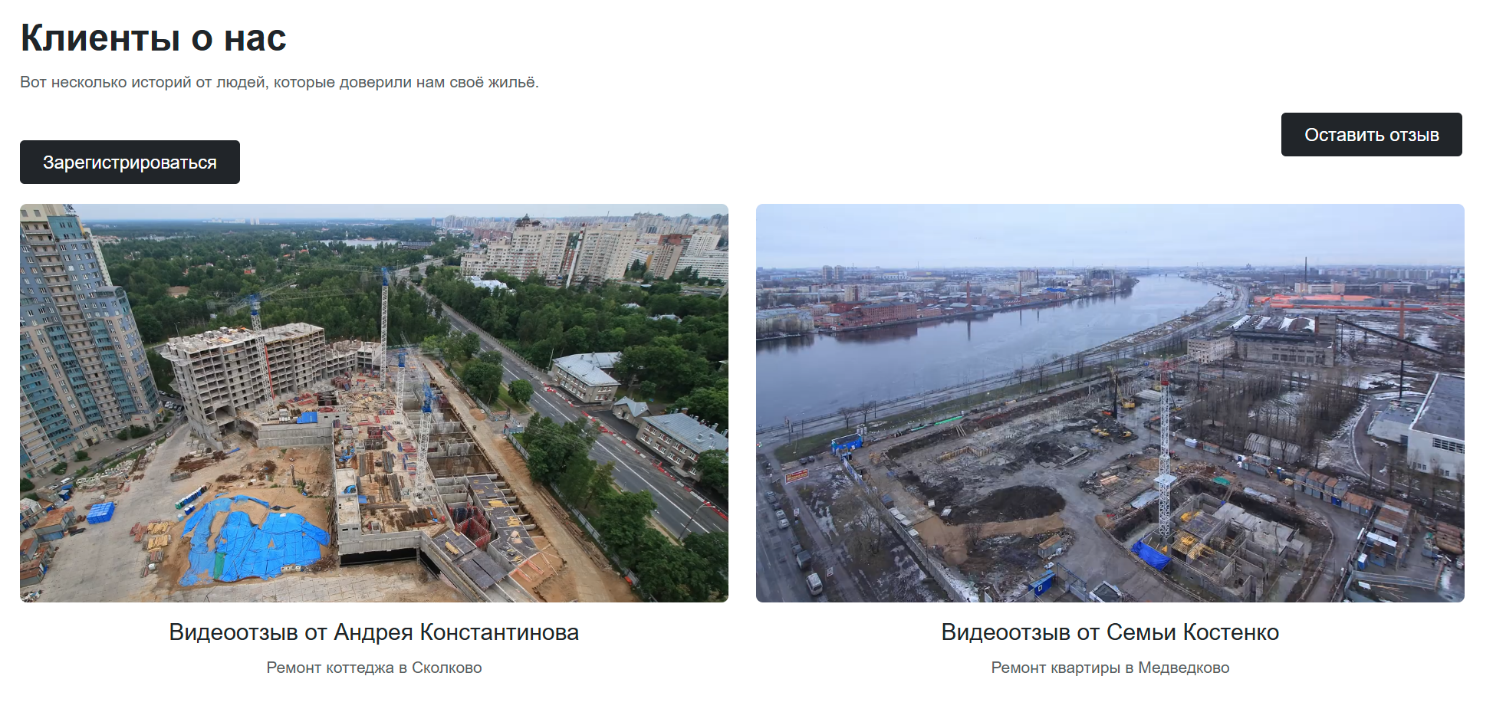


Рисунок 21 – Блок «Клиенты о нас»

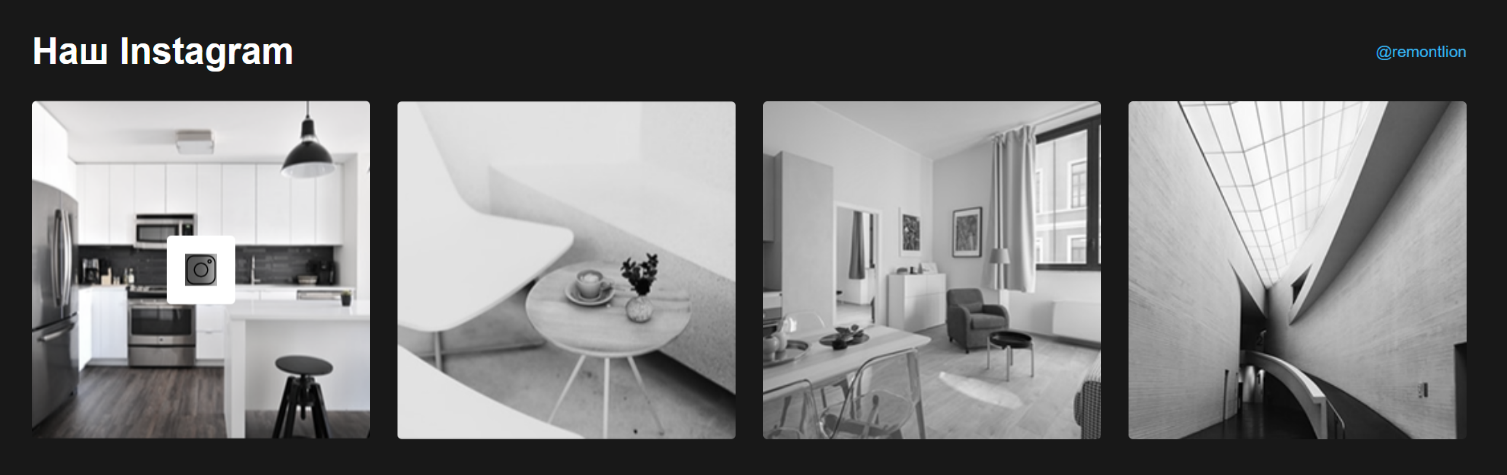
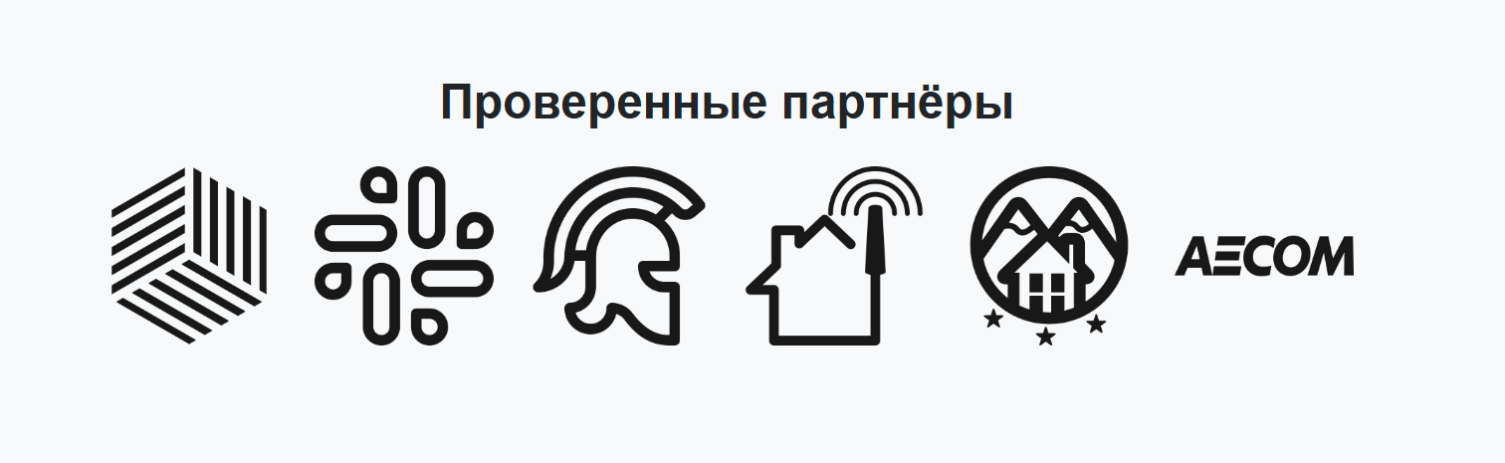


Рисунок 22 – Блок «Наша Instagram»

Рисунок 23 – Блок «Проверенные партнеры»

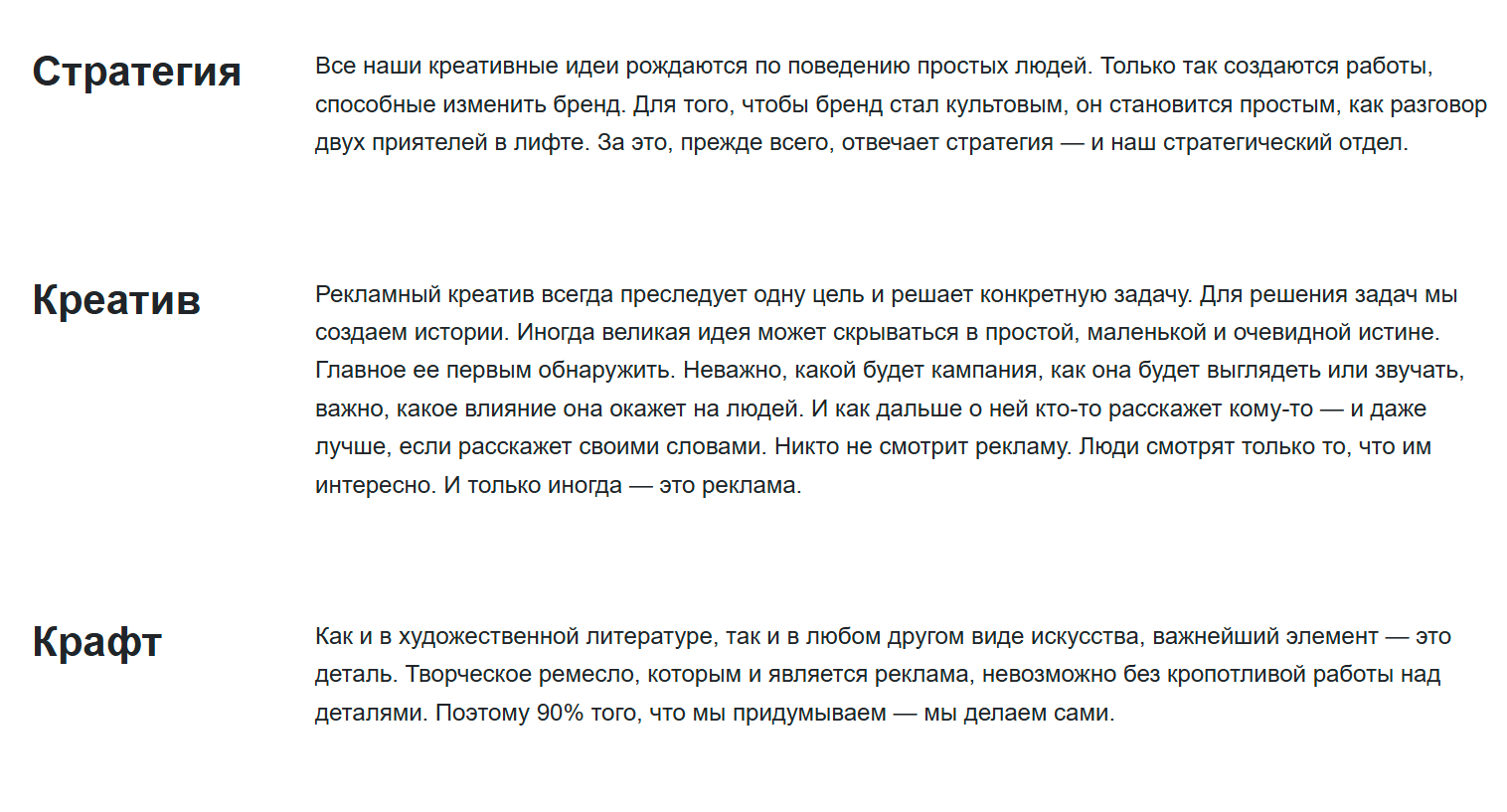
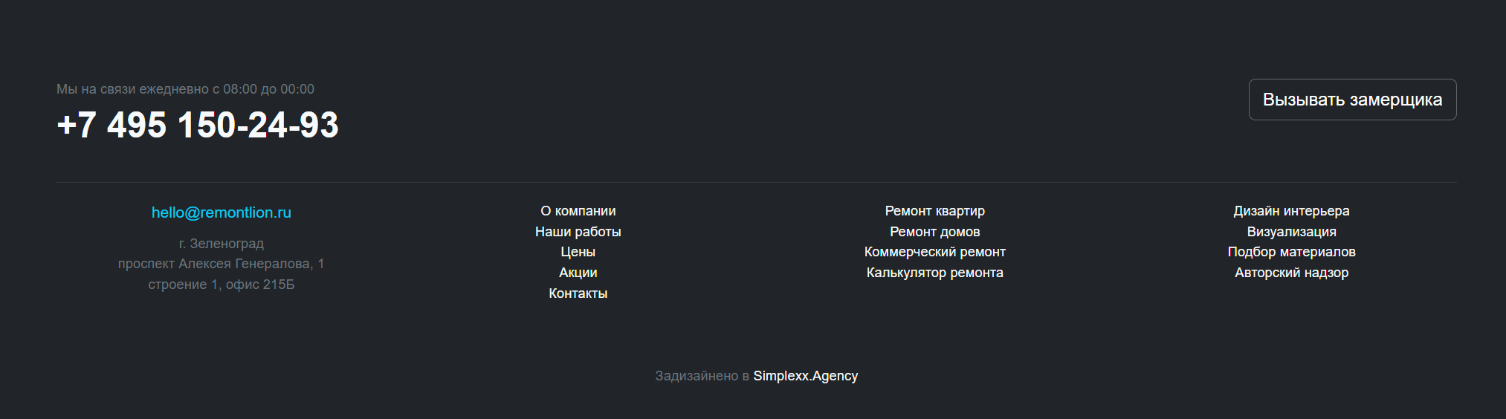


Рисунок 24 – Блок «Основы нашей работы»

Рисунок 25 – Блок «Подвал сайта»

* 1. **Тест-кейсы**

При разработке приложения важным этапом является тестирование, направленное на проверку стабильности, корректности работы функционала и соответствия требованиям. Для этого составляются тест-кейсы — документы, описывающие последовательность действий для проверки различных элементов сайта.

Использование тест-кейсов позволяет стандартизировать процесс тестирования, обеспечить единообразие проверок и снизить вероятность пропуска ошибок. В рамках проекта тест-кейсы охватывают как основные пользовательские сценарии, так и отдельные элементы интерфейса.

Готовые тест-кейсы приведены в Приложении Б.

1. **Реализация**
   1. **Руководство программиста**
      1. **Описание верстки**

Программный продукт разработан с использованием HTML и CSS, а также подключённого JavaScript. В основной шаблон главной страницы были внедрены динамические элементы CSS и интерактивные компоненты. Для функциональности был написан код на JavaScript.

Вёрстка страницы выполнена на основе макета из Figma. Основным методом адаптации элементов выступала flex-методология, а также применялись блоки и элементы Bootstrap 5.

Шапка сайта для десктопной версии реализована с использованием flex-методики. Пример строк кода представлен ниже:

<section class="testimonials py-5">

<div class="container">

<!-- Заголовок и подзаголовок, выровненные влево -->

<div class="mb-4 text-start">

<h2 class="fw-bold hover-grow">Клиенты о нас</h2>

<p class="text-muted">Вот несколько историй от людей, которые доверили нам своё жильё.</p>

</div>

<div class="row g-4">

<div class="review-button-wrapper">

<a href="forms/otz.html" class="review-button-1">Оставить отзыв</a>

<a href="forms/regist.html" class="review-button-2">Зарегистрироваться</a>

</div>

<!-- Видео 1 -->

<div class="col-md-6">

<div class="video-wrapper">

<video class="video-player w-100 rounded">

<source src="video/video1.mp4" type="video/mp4">

Ваш браузер не поддерживает видео.

</video>

</div>

<h5 class="mt-2 hover-grow">Видеоотзыв от Андрея Константинова</h5>

<p class="text-muted">Ремонт коттеджа в Сколково</p>

</div>

<!-- Видео 2 -->

<div class="col-md-6">

<div class="video-wrapper">

<video class="video-player w-100 rounded">

<source src="video/video2.mp4" type="video/mp4">

Ваш браузер не поддерживает видео.

</video>

</div>

<h5 class="mt-2 hover-grow">Видеоотзыв от Семьи Костенко</h5>

<p class="text-muted">Ремонт квартиры в Медведково</p>

</div>

</div>

</div>

</section>

Пример CSS для flex:

.main-nav .container {

display: flex;

align-items: center;

justify-content: space-between;

}

Был использован CSS3 для создания стилей и дизайна сайта. Пример кода на CSS:

.text-section h3 {

font-size: 1.8rem;

}

.text-section p {

font-size: 1rem;

line-height: 1.6;

max-width: 800px;

}

.section-container {

width: 90%;

max-width: 1200px;

margin: 60px auto;

display: flex;

flex-direction: column;

gap: 60px;

font-family: 'Arial', sans-serif;

}

.content-block {

display: flex;

gap: 40px;

align-items: flex-start;

}

.title {

min-width: 150px;

font-size: 28px;

font-weight: 700;

transition: transform 0.3s ease;

}

Были использованы медиа запросы в CSS3 для создания адаптивности сайта. При экране меньше, чем 1400px и больше 820px – адаптивный сайт для ПК с квадратным разрешением. При экране с разрешением меньше, чем 820px и больше 768px – адаптивный сайт для больших планшетов. При экране с разрешением меньше, чем 768px – адаптивный сайт для небольших планшетов и смартфонов. Пример кода с медиа-запросами представлен ниже:

@media (max-width: 768px) {

.top-bar {

display: none;

}

.main-nav {

padding: 10px 0;

}

.main-nav .container {

display: flex;

align-items: center;

justify-content: space-between;

}

.logo-block span {

font-size: 16px;

}

.nav-links,

.btn.btn-dark {

display: none !important;

}

.btn.d-lg-none {

display: flex !important;

align-items: center;

gap: 10px;

}

.btn.d-lg-none .navbar-toggler-icon {

display: inline-block;

width: 24px;

height: 2px;

background-color: #000;

box-shadow: 0 6px 0 0 #000, 0 12px 0 0 #000;

content: '';

}

.btn.d-lg-none svg {

font-size: 20px;

}

.offcanvas-body a {

font-size: 16px;

font-weight: 500;

text-decoration: none;

color: #000;

padding: 5px 0;

}

}

* + 1. **Динамические элементы на CSS**

Первоначальная стилизация сайта осуществлялась по макету, а затем по требованиям учебной практики.

На странице сайта были реализованы такие динамические элементы как:

* hover-эффекты к изображениям:

.insta-card:hover,

.insta-card:hover img {

transform: translateY(-4px);

filter: grayscale(0%);

}

* Анимация кнопок:

.top-bar a,

.top-right-links a {

color: #8a8a8a;

text-decoration: none;

transition: color 0.4s ease-in-out, transform 0.4s ease-in-out;

}

a:hover{

transition: transform 0.3s ease-in-out, font-size 0.3s ease-in-out;

transform: scale(1.1);

box-shadow: 0 4px 35px rgba(181, 181, 181, 0.1);

}

* Использование теней к блокам:

a:hover{

transition: transform 0.3s ease-in-out, font-size 0.3s ease-in-out;

transform: scale(1.1);

box-shadow: 0 4px 35px rgba(181, 181, 181, 0.1);

}

* Использование градиентов к иконкам:

.insta-card.first-card .insta-overlay img {

width: 28px;

background: linear-gradient(90deg, rgba(131, 58, 180, 1) 0%, rgba(253, 29, 29, 1) 50%, rgba(252, 176, 69, 1) 100%); /\* иконка \*/

}

* + 1. **Элементы на JavaScript**

На странице были использованы такие элементы элементы JavaScript, как:

* Анимация счетчика при появлении блока:

document.addEventListener('DOMContentLoaded', () => {

const numbers = document.querySelectorAll('.benefit-number');

let animated = false; // чтобы анимация запускалась только один раз

const animateValue = (el, target, duration = 1500) => {

const start = 0;

const startTime = performance.now();

function update(currentTime) {

const progress = Math.min((currentTime - startTime) / duration, 1);

el.textContent = Math.floor(progress \* (target - start) + start);

if (progress < 1) requestAnimationFrame(update);

else el.textContent = target; // гарантируем конечное значение

}

requestAnimationFrame(update);

};

const observer = new IntersectionObserver(entries => {

entries.forEach(entry => {

if (entry.isIntersecting && !animated) {

numbers.forEach(num => {

animateValue(num, parseInt(num.dataset.target, 10));

});

animated = true;

observer.disconnect(); // больше не нужно отслеживать

}

});

}, { threshold: 0.4 }); // 40% блока в видимой области

observer.observe(document.querySelector('.benefits-section'));

});

* Подключено при помощи строк:

<script src="js/script.js"></script>

* Переключение слайдера:

document.querySelectorAll('.top-nav .nav-link').forEach(link => {

link.addEventListener('click', function () {

document.querySelectorAll('.top-nav .nav-link').forEach(el => el.classList.remove('active'));

this.classList.add('active');

});

});

const slides = document.querySelectorAll('.slide');

const prevBtn = document.querySelector('.prev-btn');

const nextBtn = document.querySelector('.next-btn');

const counter = document.querySelector('.slider-counter');

let currentIndex = 0;

function showSlide(index) {

const totalSlides = slides.length;

const slidesContainer = document.querySelector('.slides');

const offset = -index \* 100;

slidesContainer.style.transform = `translateX(${offset}%)`;

counter.textContent = `0${index + 1} / 0${totalSlides}`;

}

nextBtn.addEventListener('click', () => {

currentIndex = (currentIndex + 1) % slides.length;

showSlide(currentIndex);

});

prevBtn.addEventListener('click', () => {

currentIndex = (currentIndex - 1 + slides.length) % slides.length;

showSlide(currentIndex);

});

Скрипт подключен с помощью строки:

<script src="js/script.js"></script>

* Обработчик клика для всех видео:

document.querySelectorAll('.video-player').forEach(video => {

video.addEventListener('click', () => {

if (video.paused) {

video.play();

} else {

video.pause();

}

});

});

document.querySelector('.btn-outline-light').addEventListener('click', function () {

alert("Спасибо! Наш замерщик свяжется с вами в ближайшее время.");

});

document.addEventListener('DOMContentLoaded', () => {

const favicon = document.querySelector('#favicon');

if (favicon) {

favicon.addEventListener('click', () => {

location.reload(); // Обновление страницы

});

}

});

Скрипт подключен с помощью строки:

<script src="js/script.js"></script>

В проекте была также была использована библиотека Bootstrap версии 5.3.3. Файл библиотеки был подключен с помощью строки:

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>

При помощи скриптов Bootstrap были созданы такие элементы как: слайдер, карусель, сетка, бургер, таб-меню, кнопки, списки. Пример кода для подключения списка:

<div class="col-md-3 mb-3">

<ul class="list-unstyled small">

<li><a href="#" class="text-light text-decoration-none">О компании</a></li>

<li><a href="#" class="text-light text-decoration-none">Наши работы</a></li>

<li><a href="#" class="text-light text-decoration-none">Цены</a></li>

<li><a href="#" class="text-light text-decoration-none">Акции</a></li>

<li><a href="#" class="text-light text-decoration-none">Контакты</a></li>

</ul>

</div>

* 1. **Спецификация файлов проекта**

Спецификация файлов проекта представлена в таблице 1.

Таблица 1 – Спецификация файлов

|  |  |  |
| --- | --- | --- |
| Папка | Файлы | Назначение |
| - | index.html | Код проекта |
| forms | calc.html, otz.html, regist.html, zamer.html | Код форм |
| css | style.css | Стили, анимации и медиа-запросы |
| globals.css | Подключение шрифтов |
| bootstrap.min.css | Библиотека Bootstrap |
| font | \*.eot, \*.woff2, \*.woff, \*.ttf | Шрифты для текста |
| video | \*.mp4 | Видео для сайта |
| img | \*.png, \*.svg, \*ico | Изображения для сайта |
| js | script.js | Валидация полей форм |
| Bootstrap.bundle.min.js | Библиотека Bootstrap |

1. **Тестирование**

В период с 3 по 6 июня происходил процесс тестирования главной страницы сайта «Ремонт квартир». Успешно было пройдено 100% тест-кейсов крайне высокого приоритета, 100% тест-кейсов высокого приоритета, 84% тест-кейсов среднего приоритета, 100% тест-кейсов низкого приоритета,

Команда тестировщиков описана в таблице 2.

Таблица 2 – Команда тестировщиков

|  |  |  |
| --- | --- | --- |
| ФИО | Должность | Роль |
| Ганисевский Александр Сергеевич | Разработчик, тестировщик | Ответственный за тестирование главной страницы сайта |

Тестирование производилось под браузер Microsoft Edge. Все тест-кейсы выполнялись вручную. Тестирование показало высокую стабильность функциональности (обнаружен один дефект с важностью «средняя»). Повторное тестирование показало высокий прирост качества (исправлено 100% обнаруженных ранее дефектов). Расписание проведения тестирования представлено в таблице 3.

Таблица 3 – Расписание работ над проектом

|  |  |  |  |
| --- | --- | --- | --- |
| ФИО | Дата | Деятельность | Продолжительность, ч |
| Ганисевский Александр Сергеевич | 03.06.2025 | Разработка тест-кейсов | 3 |
| Ганисевский Александр Сергеевич | 03.06.2025 | Тестирование главной страницы сайта | 2 |
| Ганисевский Александр Сергеевич | 04.06.2025 | Анализ выполнения тест-кейсов | 1 |
| Ганисевский Александр Сергеевич | 04.06.2025 | Тестирование главной страницы сайта. Проведение регрессивного тестирования | 3 |
| Ганисевский Александр Сергеевич | 06.06.2025 | Составление отчета о результатах тестирования | 1 |

Разработанные тест-кейсы и результаты тестирования представлены в приложении Б таблица Б.1. Статистика по разработанным тест-кейсам представлена в таблице 4.

Таблица 4 – Статистика по разработанным тест-кейсам

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Статус | Количество тест-кейсов | Приоритет | | | | |
| Крайне низкий | Низкий | Средний | Высокий | Крайне высокий |
| Пройдено | 23 | 0 | 2 | 6 | 11 | 4 |
| Не пройдено | 1 | 0 | 0 | 1 | 0 | 0 |
| Заблокировано | 0 | 0 | 0 | 0 | 0 | 0 |
| Не протестировано | 0 | 0 | 0 | 0 | 0 | 0 |

В таблице 5 представлена статистика по всем дефектам.

Таблица 5 – Статистика по выявленным ошибкам

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Статус | Количество ошибок | Важность | | | |
| Низкая | Средняя | Высокая | Критическая |
| Найдено | 1 | 0 | 1 | 0 | 0 |
| Исправлено | 1 | 0 | 0 | 1 | 0 |
| Проверено | 1 | 0 | 0 | 1 | 0 |
| Открыть заново | 0 | 0 | 0 | 0 | 0 |
| Отклонено | 0 | 0 | 0 | 0 | 0 |

Список найденных дефектов представлен в таблице 6.

Таблица 6 – Список найденных дефектов

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор | Важность | Описание | Статус |
| D\_01 | Средний | Поля не очищаются после закрытия popup | Исправлено |

1. **Применение**
   1. **Размещение на GitHub**

Данный учебный проект размещается на GitHub в репозитории WEB по ссылке: https://github.com/midland73/LionGroup

Чтобы открыть сайт перейдите по ссылке: https://midland73.github.io/LionGroup/

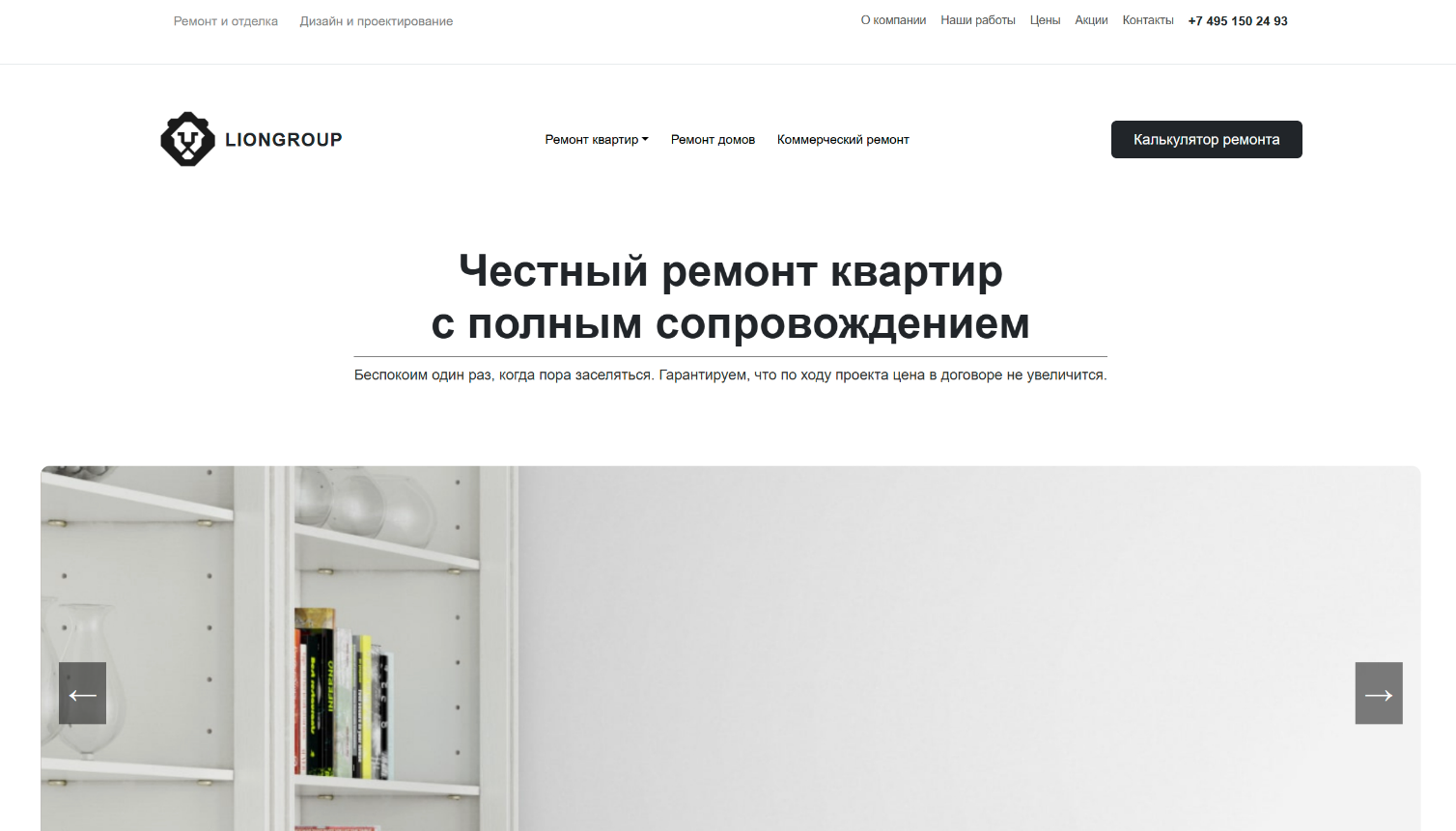
Снимок экрана главной страницы сайта представлен на рисунке 26.

Рисунок 26 – Главная страница сайта

**Заключение**

Целью прохождения учебной практики стало создание полноценного веб-шаблона на основе дизайна, представленного в макете Figma. Выполнение этого задания позволило не только закрепить теоретические знания, полученные в рамках курса «Программные средства создания Internet-приложений», но и расширить представление о современных технологиях веб-разработки, а также освоить новые инструменты и подходы к верстке.

Особое внимание в процессе разработки уделялось практическому применению таких навыков, как работа с HTML, CSS и JavaScript. Создание веб-шаблона требовало детального подхода, включая адаптивную вёрстку для различных устройств и оптимизацию скорости загрузки сайта. На завершающем этапе проводилось тестирование, которое позволило выявить и устранить мелкие недочёты. Это улучшило качество итогового продукта, сделав его более стабильным, удобным и привлекательным для пользователей.

Итогом работы стал проект – онлайн-платформа для предоставления услуг по ремонту квартир. Платформа объединяет профессиональных исполнителей и заказчиков, предоставляя удобные возможности для взаимодействия. Пользователи сайта могут ознакомиться с перечнем услуг, получить консультацию, изучить портфолио исполнителей и заказать ремонтные работы, исходя из своих потребностей и предпочтений.

Сайт был разработан с применением современных технологий, что позволило создать удобный и функциональный инструмент для пользователей. Простой интерфейс, современный дизайн и высокая скорость загрузки делают платформу конкурентоспособной и востребованной на рынке. Проект полностью отвечает актуальным требованиям: удобная навигация, визуальная привлекательность и функциональность обеспечивают положительный пользовательский опыт.

Учебная практика стала важным этапом профессионального становления, позволив реализовать реальные задачи и укрепить навыки разработки. Созданный веб-шаблон представляет собой качественное и функциональное решение, способное улучшить взаимодействие между заказчиками и исполнителями, а также способствовать развитию сферы услуг по ремонту квартир.

**Список использованных источников**

1. Шпаргалка по Flexbox: свойства, примеры использования и бесплатные тренажёры [Электронный ресурс] – Режим доступа: https://skillbox.ru/media/code/shpargalka-po-flexbox-svoystva-primery-ispolzovaniya-i-besplatnye-trenazhyery/ – Дата доступа: 29.05.2025.
2. Подробный видео курс по Bootstrap 5 на русском языке, как создать сайт на Bootstrap 5 [Электронный ресурс] – Режим доступа: https://www.youtube.com/watch?v=CvMxvb2D8Iw&ab\_channel=ITDoctor – Дата доступа: 31.05.2025.
3. Introduction [Электронный ресурс] – Режим доступа: https://getbootstrap.com/docs/5.0/getting-started/introduction/ – Дата доступа: 31.05.2025.
4. Интерактивный онлайн-курсы HTML Academy [Электронный ресурс] – Режим доступа: https://htmlacademy.ru/ – Дата доступа: 03.06.2025.
5. Интерактивный курс по JavaScript для начинающих [Электронный ресурс] – Режим доступа: https://learn.javascript.ru/ – Дата доступа: 04.06.2025.