**Введение**

В современном мире, где веб-сайты играют ключевую роль в предоставлении информации и взаимодействии с пользователями, особую значимость приобретает умение создавать удобные и привлекательные интерфейсы. Научиться вёрстке и разработке front-end части для сайта – это важный шаг к профессиональному развитию в сфере веб-разработки. Этот проект предоставляет уникальный опыт в создании функциональных и визуально привлекательных интерфейсов, что способствует развитию навыков адаптивной вёрстки, работы с CSS-фреймворками и улучшению понимания UX/UI-дизайна.

Темой проекта является разработка front-end части главной страницы сайта «Дизайн-проекты».

Целью проекта является разработка front-end части главной страницы сайта, что позволит углубить свои навыки вёрстки и создания пользовательских интерфейсов.

Далее приведено краткое описание разделов пояснительной записки.

Раздел «Анализ задачи» содержит описание постановки задачи и инструментов разработки.

В разделе «Проектирование задачи» описаны стайлгайд, цветовая палитра сайта, типографика, навигация сайта, использованные иконки, кнопки и поля ввода, компоненты и пользовательские формы.

«Реализация» – это раздел, в котором описывается руководство программитса и спецификация файлов проекта.

В четвертом разделе «Тестирование» приведено описание расписания проведения тестирования, а также статистика по выявленным ошибкам.

В разделе «Применение» описан способ для того, чтобы открыть сайт.

В заключении подведены итоги по разработке front-end части главной страницы сайта «дизайн-проекты».

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

В «Приложении А» будет приведена модульная сетка. Для декстопа, планшета и мобильного устройства

**1 Анализ задачи**

**1.1 Постановка задачи**

На учебную практику по WEB − программированию была поставлена задача по вёрстке шаблона по макету из Figma на тему: «Дизайн-проекты».

Требуется сверстать главную страницу сайта по макету, предоставленному в виде figma-ссылки:

<https://www.figma.com/design/JyeW0zGHOBi0dEAZsS2vdA/gid.group-%D0%A4%D0%B8%D0%BD%D0%B0%D0%BB-(Copy)?node-id=0-1>.

Главная страница должна быть свёрстана с учётом просмотра её на различных устройствах с разными разрешениями экрана, то есть иметь адаптивную вёрстку для просмотра на компьютере при различных разрешениях (ширина по умолчанию − 1320px, макет соответствует ширине 1281px), а также на мобильном устройстве (ширина экрана от 347px до 836px) и планшете (ширина экрана от 837px до 1280px, ширина по умолчанию 880px).

Также должен быть составлен документ, который содержит в себе список элементов и правил, подходящих под стилистику конкретного сайта – стайлгайд. Стайлгайд – это справка с набором стандартов и требований, обязательных к соблюдению. Их необходимо использовать в процессе оформления сайта. Ключевое назначение – создание единого стилистического и оформительнского однообразия. На основании данных из Figma стайлгайд должен включать в себя следующие разделы:

* логотип;
* цветовая палитра;
* сетка и отступы;
* навигация, меню;
* иконки;
* кнопки, поля ввода;
* компоненты;
* пользовательские формы;
* модальные окна, алерты и т.д.

После завершения составления стайлгайда и вёрстки макета необходимо выполнить стилизацию с помощью CSS отдельных элементов макета в соответствии со следующими критериями:

* hover-эффекты;
* оформление гиперссылок;
* тени;
* градиенты;
* оформление изображений;
* оформление(стилизация) пользовательских форм;
* трансформации;
* анимация и переходы;
* декоративные элементы;
* любые другие эффекты;

Далее необходимо добавить в свёрстанный шаблон главной страницы интерактивные, динамические элементы. При их разработке необходимо писать код на JavaScript и использовать библиотеки JQuery, JQuery UI в соответствии со следующими критериями:

* работа с окнами сообщений.
* создание диначеского меню и организация навигации;
* слайдеры;
* работа с изображениями через JS;
* использование библиотеки JQuery;
* использование библиотеки JQuery UI;
* использование любых возможных способ JS+библиотеки.

**1.2 Инструменты разработки**

Для вёрстки данного шаблона макета выбрана среда Microsoft Visual Studio со следующими установленными расширениями: Auto Raname Tag, Russian Language, HTML CSS Support, HTMLHint, Live Server, CSS Peek, CSS Navigation, Prettier, Mithrill Emmet. А также для дальнейшей работы с проектом потребуется Figma и GitHub.

Microsoft Visual Studio – редактор исходного кода. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб и облачных приложений. Имеет широкие возможности для кастомизации: пользовательские темы, сочетания клавиш и файлы конфигурации.

Figma – онлайн-сервис для разработки интерфейсов и прототипирования с возможностью организации совместной работы в режиме реального времени.

GitHub – крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки.

При вёрстке данного шаблона макета будет использован персональный компьютер со следующими характеристиками:

* процессор AMD Ryzen 5 3550H with Radeon Vega ;
* объём оперативной памяти 8.00 GB;
* объём места на жестком диске 512 GB;
* ОС Windows 10 Домашняя.

**2 Проектирование**

**2.1 Стайлгайд**

**2.1.1 Логотип и сетка**

Логотип – графический знак, эмблема или символ, используемый территориальными образованиями, коммерческими предприятиями, организации и частными лицами для повышения узнаваемости и распозноваемости в социуме. Логотип, представленный на Рисунке 1, представляет собой название сущности, которую он идентифицирует, в виде стилизованных букв и/или идеограммы.

Изображение выглядит как текст, Шрифт, Графика, логотип

Автоматически созданное описание

Рисунок 1 – Логотип

Модульная сетка сайта для персонального компьютера (ширина 1920px), планшета (ширина 880px), мобильного телефона (ширина 480px) представлена в приложении А.

**2.1.2 Цветовая палитра**

Таблица 1-Цветовая палитра сайта

|  |  |
| --- | --- |
| Элемент | Цвет |
| **1** | **2** |
| Основной фон | #FFFFFF |
| Фон шапки сайта | rgba(0, 0, 0, 0.4) |
| Фон кнопок на сайте | #00000  #FFFFF  линейный градиент(90deg, #443356, #2575fc)  rgba(0, 0, 0, 1) |
| Преимущественный текст на странице | #000000  #FFFFFF |
| Цвет текста заголков | #000000 |
| Цвет input полей | transparent |

**2.1.3 Типографика**

Типографика – сильнейший инструмент для выражения посыла в веб-дизайне. С его помощью вы можете объединить текстовую и визуальную составляющие, что поможет вам достучаться до посетителя. Типографика для шаблона макета «Органические продукты» представлена в таблице 2.

Таблица 2-Шрифты

|  |  |  |
| --- | --- | --- |
| Элемент | Шрифт | Размер |
| **1** | **2** | **3** |
| Текст заголовка h1 (desktop) | Roboto | 80px; |
| Текст заголовка h1 (planshet) | Roboto | 45px; |
| Текст заголовка h1 (mobile) | Roboto | 38px; |
| Текст заголовка h2 блока «contact-form» (desktop) | Roboto | calc(1em + 3vw) |
| Текст заголовка h2 блока «contact-form» (planshet) | Roboto | calc(1em + 3vw) |
| Текст заголовка h2 блока «contact-form» (mobile) | Roboto | calc(2em + 3vw) |
| Текст блока «grey» (desktop) | Roboto | 16px |
| Текст блока «grey» (planshet) | Roboto | 20px |
| Текст блока «grey» (mobile) | Roboto | 20px |
| Текст заголовка h2 блока «column» (desktop) | Roboto | calc(1em + 1.5vw) |
| Текст заголовка h2 блока «column» (planshet) | Roboto | calc(1em + 1.5vw) |
| Текст заголовка h2 блока «column» (mobile) | Roboto | calc(1em + 1.5vw) |
| Текст заголовка h2 блока «contact-form» (desktop) | Roboto | 20px |
| Текст заголовка h2 блока «contact-form» (planshet) | Roboto | 16px |
| Текст заголовка h2 блока «contact-form» (mobile) | Roboto | 18px |
| Текст параграфов блока «column» (desktop) | Roboto | calc(1em) |
| Текст заголовка h1 блока «contact-form» | Roboto | calc(1em + 1.6vw) |
|  |  |  |
| Продолжение таблицы 2 |  |  |
| **1** | **2** | **3** |
| Текст параграфов блока «column» (planshet) | Roboto | calc(1em) |
| Текст параграфов блока «column» (mobile) | Roboto | 14px |
| Текст параграфов блока «column» (desktop) | Roboto | 12px |
| Текст параграфов блока «column» (planshet) | Roboto | 16px |
| Текст параграфов блока «column» (mobile) | Roboto | calc(1em + 0.6vw) |
| Текст параграфов блока «left» (desktop) | Roboto | calc(1em + 2vw) |
| Текст параграфов блока «left» (planshet) | Roboto | calc(1em + 2vw) |
| Текст параграфов блока «left» (mobile) | Roboto | calc(1em + 2vw) |
| Текст параграфов блока «right» (desktop) | Roboto | calc(1em + 0.3vw) |
| Текст параграфов блока «right» (planshet) | Roboto | calc(1em + 0.3vw) |
| Текст параграфов блока «right» (mobile) | Roboto | calc(1em + 0.3vw) |
| Текст «Оставить заявку» (desktop) | Roboto | calc(1em) |
| Текст «Оставить заявку» (planshet) | Roboto | calc(1em) |
| Текст «Оставить заявку» (mobile) | Roboto | calc(1em) |
| Текст кнопки блока «block3-button» (desktop) | Roboto | calc(1em + 0.7vw) |
| Текст кнопки блока «block3-button» (planshet) | Roboto | calc(1em + 0.7vw) |
| Текст кнопки блока «block3-button» (mobile) | Roboto | calc(1em + 0.7vw) |
| Текст заголовка h3 блока «right-img» (desktop) | Roboto | calc(1em + 1vw) |
|  |  |  |
| Продолжение таблицы 2 |  |  |
| **1** | **2** | **3** |
| Текст заголовка h3 блока «right-img» (planshet) | Roboto | calc(1em + 1vw) |
| Текст заголовка h3 блока «right-img» (mobile) | Roboto | calc(1em + 1vw) |
| Текст input поля блока «right-img» (desktop) | Roboto | calc(1em + 0.2vw) |
| Текст input поля блока «right-img» (planshet) | Roboto | calc(1em + 0.2vw) |
| Текст input поля блока «right-img» (mobile) | Roboto | calc(1em + 0.2vw) |
| Текст заголовка h3 блока «footer-section» (desktop) | Roboto | 1.3em |
| Текст заголовка h3 блока «footer-section» (planshet) | Roboto | 1.3em |
| Текст заголовка h3 блока «footer-section» (mobile) | Roboto | 1.3em |
| Текст параграфов блока «footer-bottom» (desktop) | Roboto | 12px |
| Текст параграфов блока «footer-bottom» (planshet) | Roboto | 12px |
| Текст параграфов блока «footer-bottom» (mobile) | Roboto | 12px |

**2.1.4 Навигация, меню**

Важнейшей составляющей любого сайта является навигация и меню. Частая ошибка юзабили-сайта – это сложная или запутанная навигация, которая отталкивает клиента. Простое правило: чем проще ориентироваться на сайте, тем быстрее пользователь найдёт необходимую информацию.

Шапке сайта представлена на рисунке 2.



Рисунок 2 – Шапка сайта

Подвал данного макета, представленный на рисунке 3, имеет информационный посыл.

Изображение выглядит как текст, снимок экрана, диаграмма, карта

Автоматически созданное описание

Рисунок 3 – Подвал сайта

**2.1.5 Иконки**

Иконка – это максимально упрощённое 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, в презентациях и даже в рекламе.

Элементы блока шапки сайта представлены на рисунке 4.

Изображение выглядит как снимок экрана, Шрифт, Графика, логотип

Автоматически созданное описание

Рисунок 4 – Элементы блока шапки сайта

Элементы блока подвала сайта представлены на рисунке 5.

Изображение выглядит как символ, логотип, Шрифт, белый

Автоматически созданное описание

Рисунок 5 – Элементы блока подвала сайта

Элементы блока «photo-inst» представлены на рисунке 6.

Изображение выглядит как мебель, в помещении, стена, стол

Автоматически созданное описание

Рисунок 6 – Элементы блока «photo-inst»

**2.1.6 Кнопки и поля ввода**

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

Кнопка, представленная на рисунке 7, «.main-button» находится в блоке «.main-medium»

Изображение выглядит как снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 7 – Кнопка «.main-button»

Кнопка, представленная на рисунке 8, находится в блоке «.columnleft».

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Автоматически созданное описание

Рисунок 8 – Кнопка блока «.columnleft»

Кнопка, представленная на рисунке 9, находится в блоке «.button».

Изображение выглядит как текст, снимок экрана, Шрифт, логотип

Автоматически созданное описание

Рисунок 9 – Кнопка в блоке «.button».

Кнопка, представленная на рисунке 10, находится в блоке «.block3-button».



Рисунок 10 – Кнопка в блоке «.block3-button

Кнопка, представленная на рисунке 11, «.get» находится в блоке «.right-img».

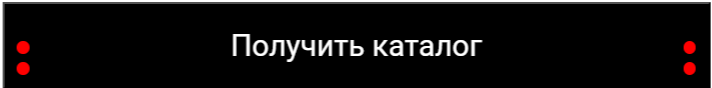


Рисунок 11 – Кнопка «.get»

Кнопка, представленная на рисунке 11, находится в блоке «.contact-form»

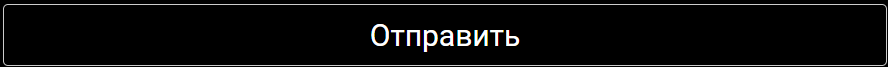


Рисунок 11 – Кнопка в блоке «.contact-form»

Кнопка, представленная на рисунке 13, «.radio» находится в блоке «.right-img».



Рисунок 13 – Кнопка «.radio»

Теперь покажем поля для ввода пользователем на сайте.

Поле ввода, представленная на рисунке 12, «.email» находится в блоке «.right-img».



Рисунок 12 – Поле ввода «.email»

Поля для ввода, представленные на рисунке 14, находятся в блоке «.form-group».



Рисунок 14 – Поля ввода в блоке «.form-group»

**2.1.7 Компоненты**

Кнопка, которая представленна на рисунке 15, выполняет функцию «Scroll» на самых вверх сайта.



Рисунок 15 – Кнопка «Scroll»

Уведомление, которое представленно на рисунке 16, выполняет функцию уведомить пользователя о новых акциях, текст меняется при каждом заходе на сайт.

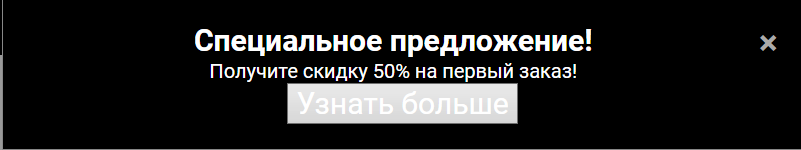


Рисунок 16 – Уведомление

Выпадающее меню, которое представленно на рисунке 17.

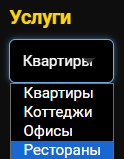


Рисунок 17 – Выпадающее меню

Анимация кнопки представлено на рисунке 18-19.

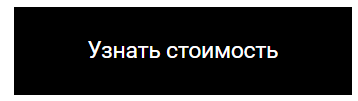


Рисунок 18 – Обычное состояние кнопки



Рисунок 19 – Анимация при наведении

Увеличение изображения на рисунке 20-21, выполняет функцию дизайна на странице.



Рисунок 20 – Обычное изображение



Рисунок 21 – Увеличение изображения при наведении

Слайдер, представленный на рисунке 22, выполняет функцию слайда мнения критиков а так же описания самих критиков.

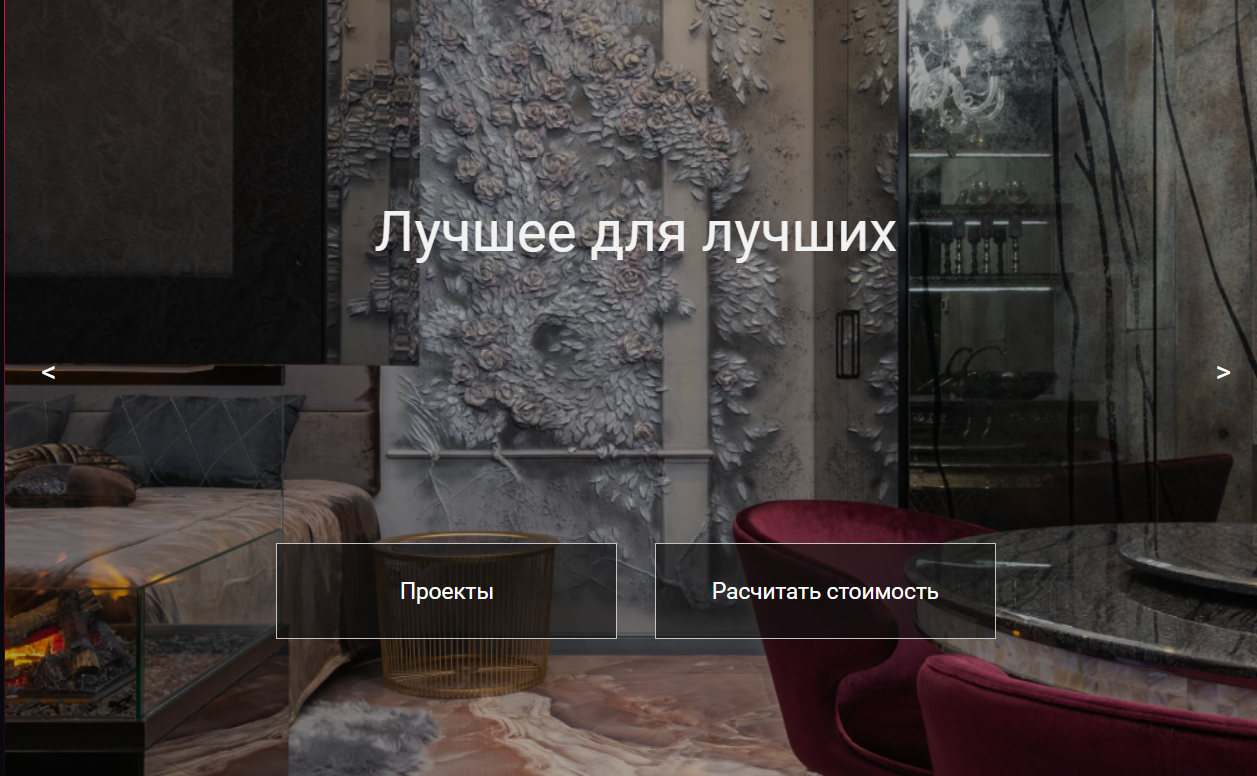


Рисунок 22 – Слайдер

Валидация почты, представленная на рисунке 23, выполняет функцию проверки правильности ввода почты в форму, в противном случае появляется сообщение об ошибке.

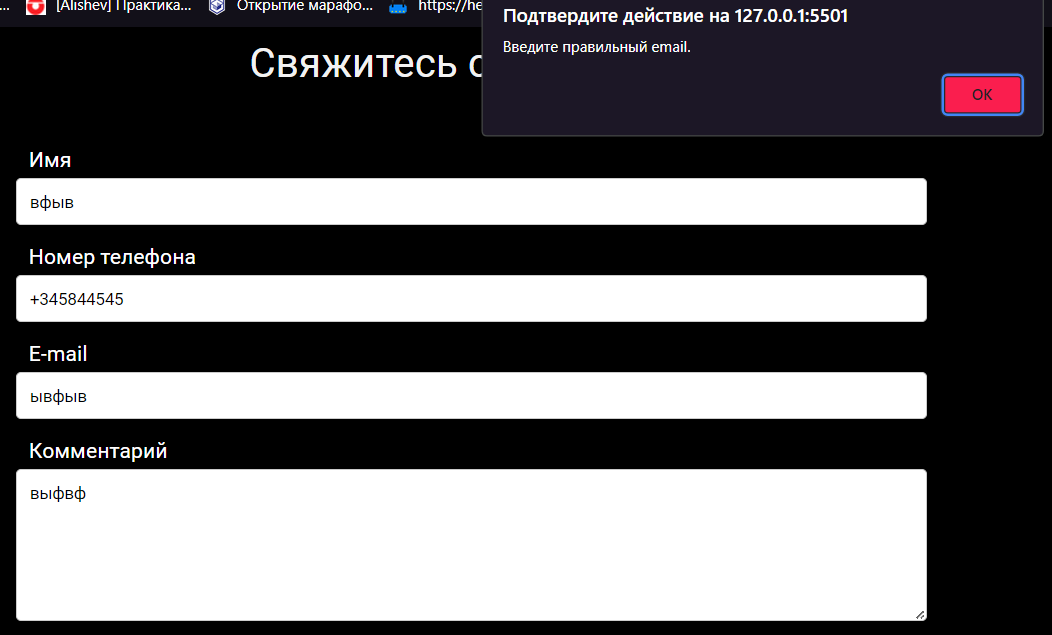


Рисунок 23 – Валидация номера телефона

Валидация номера телефона, представленная на рисунке 24, выполняет функцию проверки правильности ввода номера в форму, в противном случае появляется сообщение об ошибке.

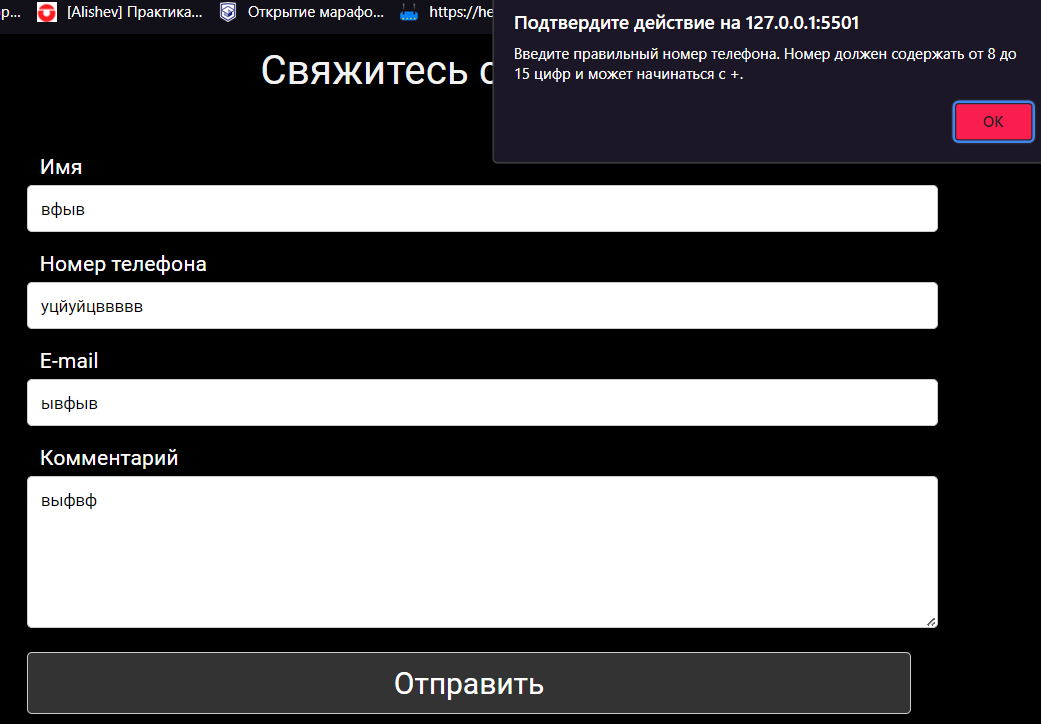


Рисунок 24 – Валидация номера телефона

**2.1.8 Пользовательские формы**

Пользовательской формой, представленной на рисунке 25, является форма «Обратной связи».

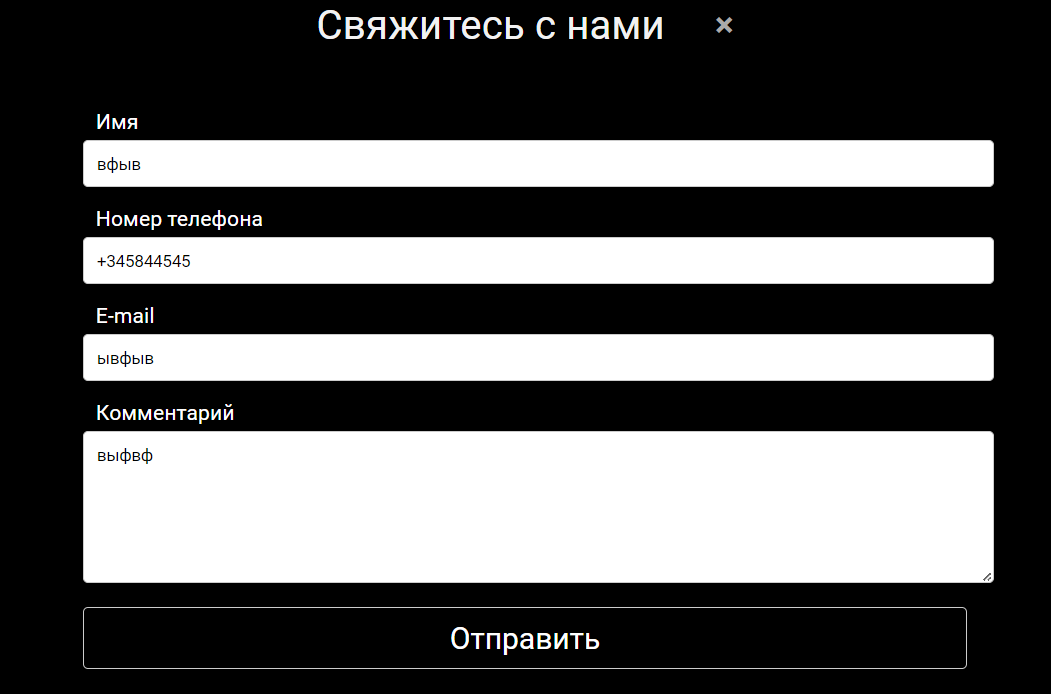


Рисунок 25 – Пользовательская форма

Форма служит для связи пользователя с сайтом.

**3 Реализация**

**3.1 Руководство программиста**

Общие требования:

Сайт должен быть разработан с помощью HTML, CSS и JS.

Сайт должен быть совместим с популярными браузерами и устройствами.

Сайт должен обеспечивать безопасность и защищенность данных клиентов.

Структура проекта:

Использовать HTML5 для создания структуры страницы.

Использовать семантические теги для улучшения доступности и поисковой оптимизации. Пример кода HTML:

<section class="some-information">

<div class="grey"> <!-- серая часть -->

<div class="column"> <!-- колонки -->

<h2>100+</h2><br>

<b>Реализованные проекты</b> <br>

<p>Работаем с любой недвижимостью – от дворцов до квартир</p>

</div>

<div class="column"><h2>10 лет</h2><br>

<b>Безупречная репутация</b> <br>

<p>Наши специалисты постоянно стажируются у лидеров строительных работ</p></div>

<div class="column"><h2>3 года</h2><br>

<b>Гарантии на все работы</b> <br>

<p>Мы внимательно изучаем рынок отделочных материалов и инновационной техники

</p></div>

<div class="column"><h2>Сроки</h2><br>

<b>Процесс</b> <br>

<p>Нужно лучшее – закажи это у лучших!

</p></div>

</div>

CSS

Использовать CSS3 для создания стилей и дизайна сайта. Пример кода CSS

/\* Кнопка закрытия \*/

.close {

color: #aaa;

float: right;

font-size: 28px;

font-weight: bold;

}

.close:hover,

.close:focus {

color: white;

background-color: #000;

text-decoration: none;

cursor: pointer;

}

.custom-select-container::after {

content: "";

position: absolute;

top: 50%;

right: 15px;

width: 0;

height: 0;

pointer-events: none;

border-left: 5px solid transparent;

border-right: 5px solid transparent;

border-top: 5px solid #333;

transform: translateY(-50%);

}

//Медиа-запрос

@media screen and (min-width: 1281px){

#accordion {

display: block;

}

.container{

position: absolute;

height: 1080px;

}

.map{

display: flex;

position: absolute;

left: 55%;

top: 0;

}

.heder .close{

position: relative;

left: 75px;

top: 35px;

width: 45px;

}

.as {

display: flex;

width: 400px;

justify-content: space-between;

}

JS Использовать JavaScript для создания интерактивных элементов и функциональности сайта.

Использовать библиотеку jQuery для упрощения написания JS-кода. Пример кода JavaScript

**//бургер меню**

document.querySelector('.p1').addEventListener('click', function (e) {

var div = document.querySelector('.shado')

div.style.display = 'flex';

div.style.flexDirection = 'column';

var dbutton = div.querySelectorAll('button');

dbutton.forEach(button => {

button.style.display = 'block';

});

})

document.querySelector('.close').addEventListener('click', function (e) {

var div = document.querySelector('.shado')

div.style.display = 'none';

div.style.flexDirection = 'column';

var dbutton = div.querySelectorAll('button');

dbutton.forEach(button => {

button.style.display = 'none';

})

});

**// уведомление**

window.addEventListener('load', () => {

setTimeout(() => {

let div = document.createElement('div');

div.classList.add('modal');

div.innerHTML = ` <div class="modal-content">

<span class="close" id=c1>&times;</span>

<h2>Специальное предложение!</h2>

<p>Получите скидку 50% на первый заказ!</p>

<button>Узнать больше</button>

</div>`;

let main = document.querySelector('.main-content');

main.appendChild(div);

div.style.display = "block";

var span = document.getElementById("c1");

span.onclick = function() {

div.style.display = "none";

}

// Когда пользователь кликает в любом месте за пределами модального окна, закрываем его

window.onclick = function(event) {

if (event.target == div) {

div.style.display = "none";

}

}

}, 5000);

})

**// карусель**

let currentIndex = 0;

const intervalTime = 3000; // Интервал в миллисекундах (3 секунды)

const carouselInner = document.querySelector('.carousel-inner');

const totalItems = document.querySelectorAll('.carousel-item').length;

function nextSlide() {

currentIndex++;

if (currentIndex >= totalItems) {

currentIndex = 0;

}

updateCarousel();

}

function updateCarousel() {

const offset = currentIndex \* -100 + '%';

carouselInner.style.transform = 'translateX(' + offset + ')';

}

**// Запуск автоматической прокрутки**

setInterval(nextSlide, intervalTime);

**// прокрутка вверх сайта**

var amountScrolled = 200;

$(window).scroll(function() {

if ( $(window).scrollTop() > amountScrolled ) {

$('a.back-to-top').fadeIn('slow');

} else {

$('a.back-to-top').fadeOut('slow');

}

});

$('a.back-to-top').click(function() {

$('html, body').animate({

scrollTop: 0

}, 700);

return false;

});

**//валидация**

let getCalatog = document.querySelector('.get');

let radio = document.querySelector('.radio');

let input = document.querySelector('.email');

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

if (input.value == '') {

alert('Поле не должно быть пустым!');

}

else if (input.value.length<=10) {alert('email должен быть не короче 10 символов!')}

else if (!radio.checked) {alert('Согласитесь на обработку данных!')}

else alert('Удачно!');

})

**//обратная связь**

let phoneB = document.querySelector('.p3');

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

let div = document.querySelector('.contact-form');

div.style.width = '100%' ;

let main = document.querySelector('.main-content');

let main\_up = document.querySelector('.main-up');

let main\_medium = document.querySelector('.main-medium');

main\_up.style.display = "none";

main\_medium.style.display = "none";

div.style.display = "flex";

var span = document.getElementById("close");

span.onclick = function() {

div.style.display = "none";

div.style.width = '0%' ;

main.style.display = "flex";

main\_up.style.display = "block";

main\_medium.style.display = "flex";

}

**// валидация**

let input1 = document.getElementById('name');

let input2 = document.getElementById('phone');

let input3 = document.getElementById('email');

let input4 = document.getElementById('comment');

let sendB = document.getElementById('submit1');

let phonePattern = /^\+?[0-9\s\-]{8,15}$/;

let emailPattern = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

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

if (input1.value === '' || input2.value === '' || input3.value === '' || input4.value === '') {

alert('Поле не должно быть пустым!');

} else if (input2.value.length < 8) {

alert('Номер должен быть больше 8 символов');}

else if (!phonePattern.test(input2.value)) {

alert('Введите правильный номер телефона. Номер должен содержать от 8 до 15 цифр и может начинаться с +.');

}else if (!emailPattern.test(input3.value)) {

alert('Введите правильный email.');

}

else if (input4.value.length < 10) {

alert('Введите не короткий комментарий!');

} else {

alert('Удачно!');

div.style.display = "none";

div.style.width = '0%' ;

main.style.display = "flex";

main\_up.style.display = "block";

main\_medium.style.display = "flex";

}

});

})

**//слайдер**

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

const prevButton = document.querySelector('.prev-button');

const nextButton = document.querySelector('.next-button');

const slides = slider.querySelectorAll('.start');

const slideCount = slides.length;

let slideIndex = 0;

**// Устанавливаем обработчики событий для кнопок**

prevButton.addEventListener('click', showPreviousSlide);

nextButton.addEventListener('click', showNextSlide);

**// Функция для показа предыдущего слайда**

function showPreviousSlide() {

slideIndex = (slideIndex - 1 + slideCount) % slideCount;

updateSlider();

}

**// Функция для показа следующего слайда**

function showNextSlide() {

slideIndex = (slideIndex + 1) % slideCount;

updateSlider();

}

**// Функция для обновления отображения слайдера**

function updateSlider() {

slides.forEach((slide, index) => {

if (index === slideIndex) {

slide.style.display = 'block';

} else {

slide.style.display = 'none';

}

});

}

**// Инициализация слайдера**

updateSlider();

<script>

$(document).ready(function(){

**// Аккордеон**

$("#accordion").accordion();

**// Подсказки**

$(document).tooltip();

});

</script>

**3.2 Спецификация файлов проекта**

Таблица 3 – Спецификация файлов проекта

|  |  |  |  |
| --- | --- | --- | --- |
| Имя корневой папки | Имя папки | Имя файла | Назначение |
| **1** | **2** | **3** | **4** |
| practica | css | 1.css | Файл стилей |
|  |  | carusel.css | Файл карусели |
|  |  | slider.css | Файл слайдера |
| practica | Img | 1.1.png | Изображение интерьера 1 |
|  |  | 1.2.png | Изображение интерьера 2 |
|  |  | 1.3.png | Изображение интерьера 3 |
|  |  | 1.4.png | Изображение интерьера 4 |
|  |  | 2.1.png | Изображение интерьера 5 |
|  |  | 2.2.png | Изображение интерьера 6 |
|  |  | 2.3.png | Изображение интерьера 7 |
|  |  | 2.4.png | Изображение интерьера 8 |
|  |  | 2.5.png | Изображение интерьера 9 |
|  |  | 2.6.png | Изображение интерьера 10 |
|  |  | 4.png | Изображение интерьера 11 |
|  |  | 5.1.png | Изображение интерьера 12 |
|  |  | Arrow1.png | Изображение стрелки вниз |
|  |  | back-to-top.png | Изображение стрелки вверх |
|  |  | galochka.png | Изображение галочки |
|  |  | icon.png | Изображение телеграмм |
|  |  | Img.footer.png | Фон для подвала |
|  |  | in11.png | Изображение footer1 |
|  |  | in22.png | Изображение footer2 |
|  |  | inst.png | Изображение инстаграмма |
|  |  | int1.png | Изображение интерьера 13 |
|  |  | palka.png | Изображение палки |
|  |  | Rectangle56.png | Изображение секции |
|  |  | tg.png | Изображение ютуб |
|  |  | xrest.png | Изображение крестика |
|  |  | 3.1.svg | Иконка для секции1 |
|  |  | 3.2.svg | Иконка для секции2 |
|  |  | 3.3.svg | Иконка для секции3 |
|  |  | 3.4.svg | Иконка для секции4 |
|  |  | 3.5.svg | Иконка для секции5 |
|  |  | 3.6.svg | Иконка для секции6 |
|  |  | Arrow4.svg | Стрелка вправо |
|  |  | grey\_block.svg | Серый квадрат |
|  |  | Rectangle17.svg | Черный квадрат |
|  |  | Vector(1).svg | Иконка телефона |
|  |  | Vector.svg | Иконка бургер-меню |
|  |  | Vector3.svg | Иконка для кнопки |
|  |  | Whiteblock.svg | Белый квадрат |
| practica | js | 1.js | Скрипты для файла |
|  |  | carusel.js | Скрипт карусели |
|  |  | slider.js | Скрипт слайдера |
|  |  | jquery-3.7.1.min.js | Скрипт библиотеки |
| practica |  | index.html | Главная страница |

**4 Тестирование**

Тестирование является одним из самых важнейших этапов при создании программного продукта.

При тестировании программного продукта, в первую очередь нужно обратить внимание на правильную работу страницы. Нужно проверить верную работу всех подключённых скриптов для правильной работы сайта.

Таким образом, основная часть ошибок и недоработок была выявлена и исправлена на этапе реализации проекта. После завершения этапа реализации, было проведено тщатильное тестирование.

Таблица 2 – Отчёт о результатах тестирования функций сайта.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название теста | Шаги | Исходная информация | Ожидаемый результат | Физический результат | Результат тестирования |
| **1** | **2** | **3** | **4** | **5** | **6** |
| 1 Появление якоря | 1 Прокрутить сайт вниз | Отсутствует | Появление pup-up | Появление pup-up | Выполнено |
| 2 Возврат в начало страницы с помощью якоря | 1 Нажать на иконку в правом нижнем углу | Отсутствует | Возврат в начало страницы | Возврат в начало страницы | Выполнено |
| 3 Проверка прокрутки карусели | 1 Прокрутить вниз сайта | Отсутствует | Карусель листается влево | Карусель листается влево | Выполнено |
| 4 Проверка появления уведомления | 1 Зайти на сайт  2 Подождать 5 секунд | Отсутствует | Появляется уведомление | Появляется уведомление | Выполнено |
| 5 Проверка анимации на кнопках | 1 Навести курсор мыши на кнопку | Отсутствует | Анимация на кнопках есть | Анимация на кнопках есть | Выполнено |
| 6 Проверка увеличения изображение | 1 Навести курсор мыши на фотографию | Отсутствует | Увеличилось изображение | Увеличилось изображение | Выполнено |

Ошибки не найдены, соотвественно, тестирование прошло успешно и программный продукт характеризуется как реализованный и готовый к работе.

**5 Применение**

Для того чтобы открыть сайт, представленный на рисунке 26, необходимо перейти по ссылке: https://jamalbenderusss.github.io/

Изображение выглядит как снимок экрана, текст, мебель, диван

Автоматически созданное описание

Рисунок 26 – Главная страница сайта

**Заключение**

Целью учебной практики являлась вёрстка шаблона макета из Figma.

В ходе реализации поставленной задачи был укреплён пройденный изученный материал по дисциплине «Программные средства Internet-приложений», а также получено множество информации о её возможностях.

После тщательного тестирования сайта были выявлены некоторые недоработки, которые были исправлены на стадии проектирования. В целом при реализации программы, были выполнены все необходимые условия. Таким образом, можно сказать, что программный продукт был реализован успешно.

**Список использованных источников**

1 Основы вёрстки сайта – [https://practicum.yandex.ru/blog/ chto-takoe-verstka-sayta/](https://practicum.yandex.ru/blog/%20chto-takoe-verstka-sayta/)

2 Подключение и выполнение JavaScript – https:// javascript.ru /tutorial/ foundation/start

3 Что должен знать верстальщик HTML, CSS - [https://medium.com/о-дизайне/верстка-для-самых-маленьких-или-что-дизайнер-должен-знать-об-html-и-css-и-где-это-найти](https://medium.com/о-дизайне/верстка-для-самых-маленьких-или-что-дизайнер-должен-знать-об-html-и-css-и-где-это-найти-261370c6aaaa)

Приложение А

«Модульная Сетка»

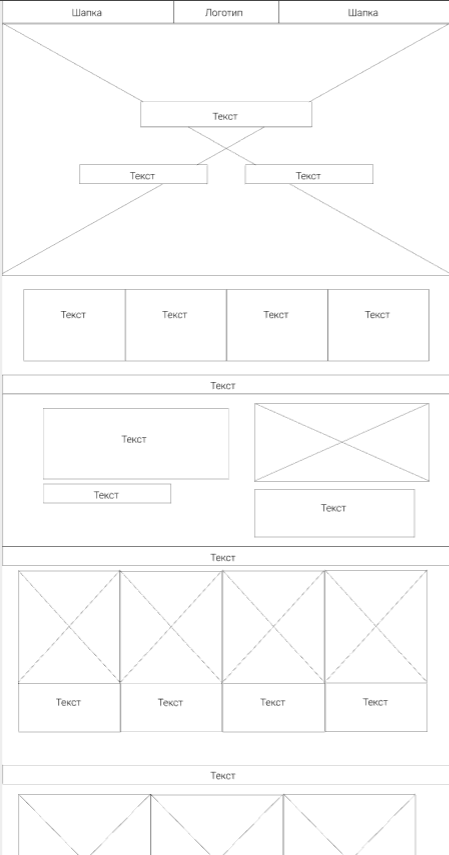


Рисунок А.1.1 – ПК версия

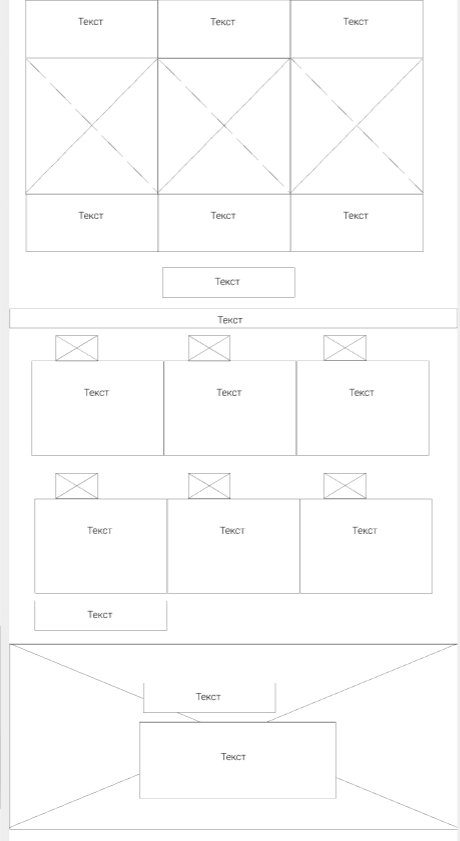


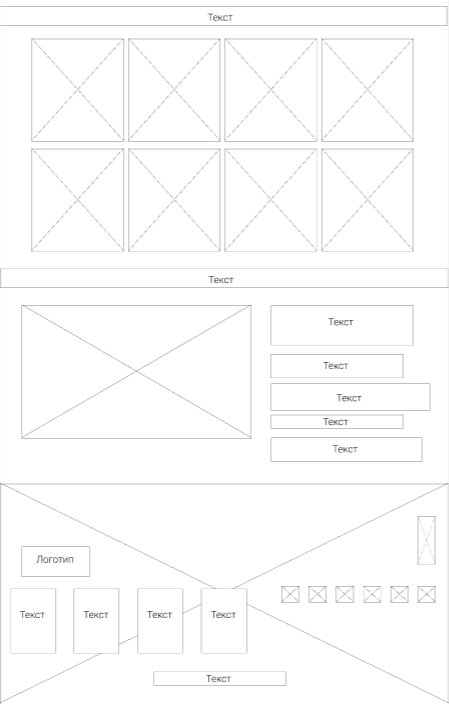
Рисунок А.1.2 – ПК версия

Рисунок А.1.3 – ПК версия

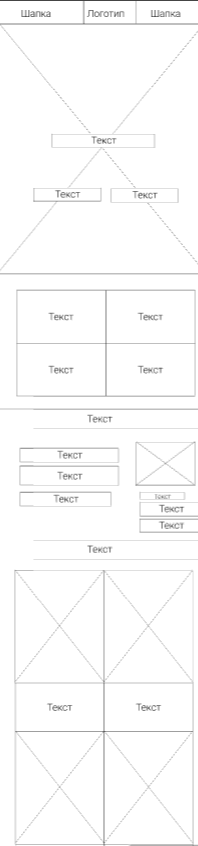
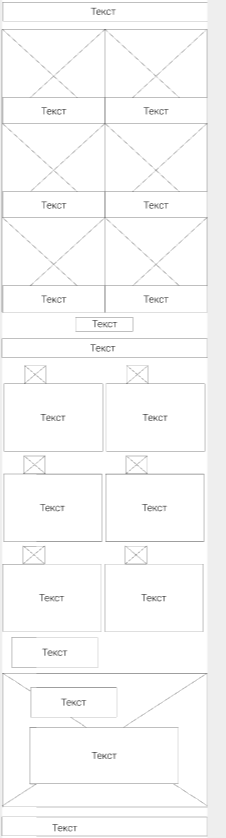


Рисунок А.2.1 – Планшет версия



Рисунок А.2.2 – Планшет версия

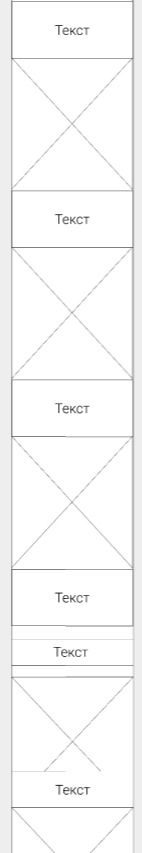


Рисунок А.3.1 – Мобильная версия

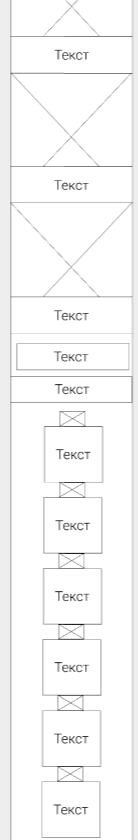


Рисунок А.3.2 – Мобильная версия