

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ	5
1.1 Постановка задачи	5
1.2 Выбор инструментов для разработки дизайна и вёрстки лендинга	5
1.3 Поиск веб-сайтов со схожей тематикой	6
1.4 Реализация дизайна лендинга	8
1.5 Вёрстка лендинга	13
ЗАКЛЮЧЕНИЕ	21
СПИСОК ИСТОЧНИКОВ	22

					СМА.509000		
Изм.	Лист	№ докум.	Подпись	Дата	Проектирование и реализация веб-сайта по теме «Лендинг для онлайн-магазина оригинальных подарков и сувениров»		
Разраб.		Семенович М.А.					
Провер.		Гурьева Н.А.					
Реценз.							
Н. Контр.							
Утверд.							
					для	Лист	Листов
					3	22	
					Учреждение образования «Полоцкий государственный университет имени Евфросинии Полоцкой» гр.22- ИТ-2		

ВВЕДЕНИЕ

В данном отчете рассматривается процесс создания лендинга для онлайн-магазина оригинальных подарков и сувениров, его визуальное оформление, вёрстка и основные технологии, используемые в веб-разработке, такие как HTML, CSS и JavaScript.

Лендинг пейдж (landing page) или просто «лендинг» — это интернет-страница, которая при помощи краткой презентации продукта побуждает пользователя совершить целевое действие [1]. Лендинги часто используются для продвижения продуктов, сбора контактных данных, регистрации на мероприятия или других целей. Их эффективность зависит от дизайна, структуры, текстового контента и действенных элементов взаимодействия. Важной чертой лендинга является его фокусировка на конкретной задаче и минималистичный дизайн, способствующий достижению этой цели.

Важной частью процесса создания является вёрстка. Вёрстка представляет собой процесс преобразования дизайн-макета в код, который браузер может понимать и отображать для пользователей. Вёрстка веб-страницы включает использование языков разметки, таких как HTML, для организации структуры страницы, включая заголовки, параграфы, изображения и ссылки. CSS используется для стилизации элементов, позволяя задавать цвета, шрифты и расположение.

Для добавления интерактивности и динамических элементов на лендинге, разработчики прибегают к JavaScript. Этот язык программирования позволяет создавать анимации, обрабатывать ввод пользователя, загружать данные с сервера без перезагрузки страницы и многое другое. JavaScript дополняет HTML и CSS, делая лендинг более функциональным и привлекательным для посетителей.

					СМА.509000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

1 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

1.1 Постановка задачи

В рамках производственной практики было принято решение уделить основное внимание процессу создания лендинга, начиная с разработки дизайна и заканчивая его вёрсткой и функциональностью с использованием HTML, CSS и JavaScript. Задачей данной практики было изучение всех этапов создания лендинга с акцентом на проектировании и разработке пользовательского интерфейса.

Первым делом разрабатывался дизайн лендинга, что позволило создать наглядные макеты страниц, определить структуру сайта и визуальное оформление. Этот этап позволяет определить цветовую палитру, шрифты, расположение элементов и общий стиль лендинга, что является фундаментом успешного дизайна.

Далее основное внимание уделялось вёрстке с использованием языков разметки HTML и стилевых таблиц CSS. Вёрстка включает в себя размещение текстовых и графических элементов, создание адаптивного дизайна для различных устройств.

Для придания лендингу интерактивности и динамической функциональности использовались средства языка программирования JavaScript. JavaScript позволяет создавать анимации, обрабатывать пользовательские действия и другие задачи, делая лендинг более привлекательным и функциональным.

1.2 Выбор инструментов для разработки дизайна и вёрстки лендинга

Для создания дизайна используется приложение Figma. Figma – это онлайн-платформа для создания макетов и прототипов сайтов и приложений. Данный инструмент будет полезен не только дизайнерам, но и команде по работе с проектом, разработчикам, так как упрощает многие моменты, в том числе и вёрстку [2]. Данное приложение обладает рядом преимуществ.

Figma достаточно просто в использовании. Интерфейс Figma интуитивно понятен и дружелюбен к пользователям, что упрощает процесс создания дизайна. Figma предоставляет широкий набор инструментов для создания векторных и растровых изображений, а также очень важным моментом является возможность использования плагинов для расширения функциональности (например, различные значки, часто используемые в дизайне).

Для написания кода на HTML, CSS и JavaScript был выбран Visual Studio Code (VS Code). VS Code представляет собой обычный текстовый

					СМА.509000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

редактор с возможностью подключения различных плагинов, что позволяет работать со всевозможными языками программирования для разработки любого IT-продукта [3].

VS Code удобен в использовании, так как имеет такие полезные преимущества как подсветка синтаксиса для лучшего понимания структуры кода и возможности в нём быстро ориентироваться, автодополнение для ускорения процесса написания кода и снижения вероятности ошибок.

1.3 Поиск веб-сайтов со схожей тематикой

В начале разработки дизайна сайта было решено найти идеи на уже имеющихся в открытом доступе веб-сайтах с такой же тематикой. Большую роль в идейном вдохновлении сыграл лендинг сервиса Rgbstore.by (адрес - <https://rgbstore.by/>). Данный веб-сайт специализируется на предоставлении товаров для продажи: неоновых вывесок и изделий из неона.

Зайдя на страницу, пользователь видит перед собой, погружённый во тьму, светящийся торшер, который отражает специализацию данного сервиса (Рисунок 1). Важной особенностью этого сайта можно считать эффект шума на заднем фоне, добавляющего необычную атмосферу всей странице. Также хорошо подобранные цвета сайта полностью соответствуют его названию и области деятельности компании. Однако, уже в самом начале сайта видно не очень грамотное расположение кнопки заказа, так как она находится в самом низу и не привлекает тем самым к себе внимания, чуть ли не обрезается в начальном положении страницы.

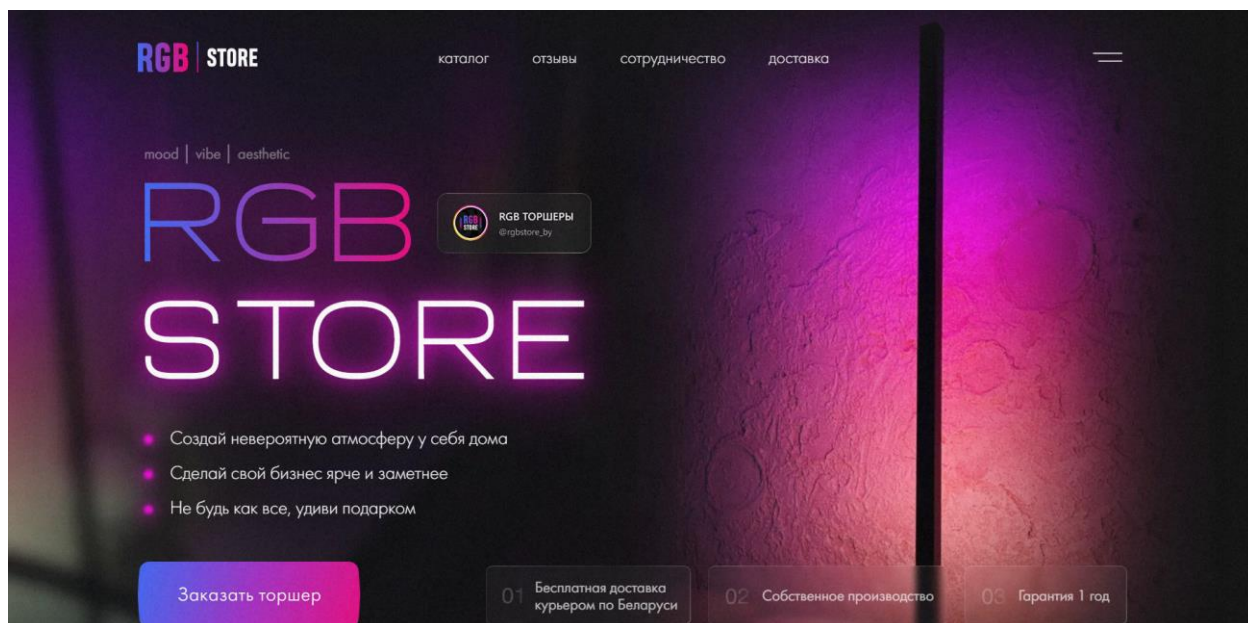


Рисунок 1 – верхняя часть страницы

Чуть ниже находится блок с главными преимуществами сервиса Rgbstore (Рисунок 2). Стоит отметить то, как хорошо подобраны цвета элементов. Они дополняют фон и другие объекты, формируя общую,

					СМА.509000	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

радующую глаз, картину. Важным атрибутом блока, как и некоторых последующих, является использование эффекта прозрачного стекла на фоне.

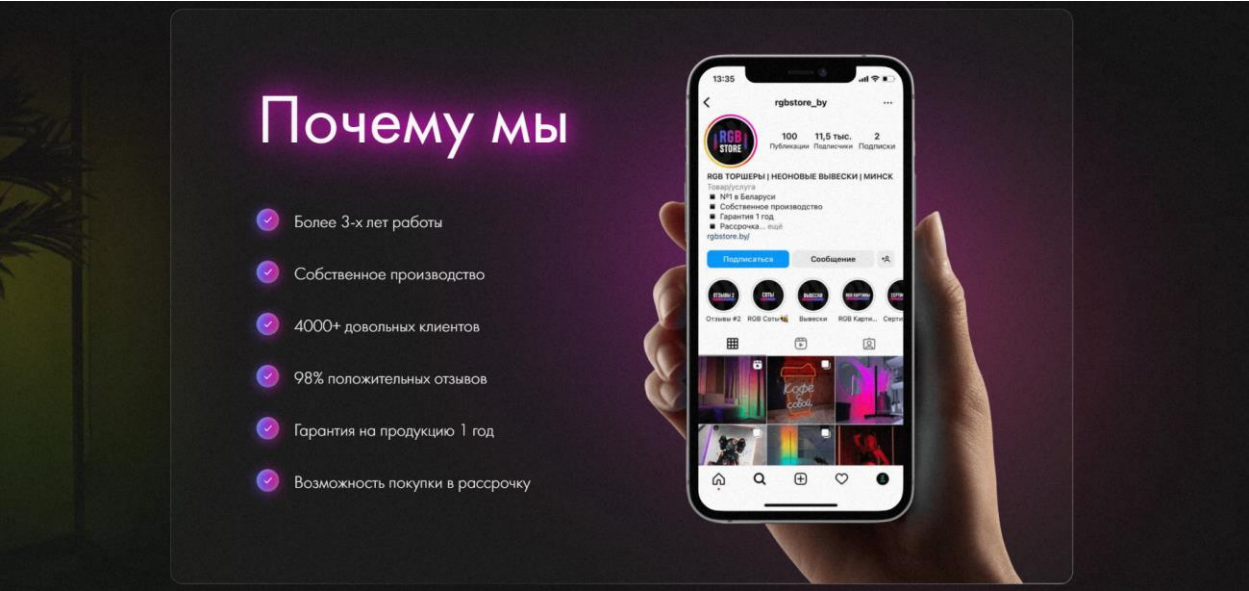


Рисунок 2 – блок преимуществ

Далее расположен блок с сертификатами (Рисунок 3). Сами по себе они оформлены хорошо, однако блоки достаточно большие, и несколько таких блоков, расположенных подряд, как в случае с лендингом Rgbstore, выглядят не так гармонично. Но, с другой стороны, весьма интересно представлена картинка подарочных сертификатов, добавляющая динамики полученному блоку.

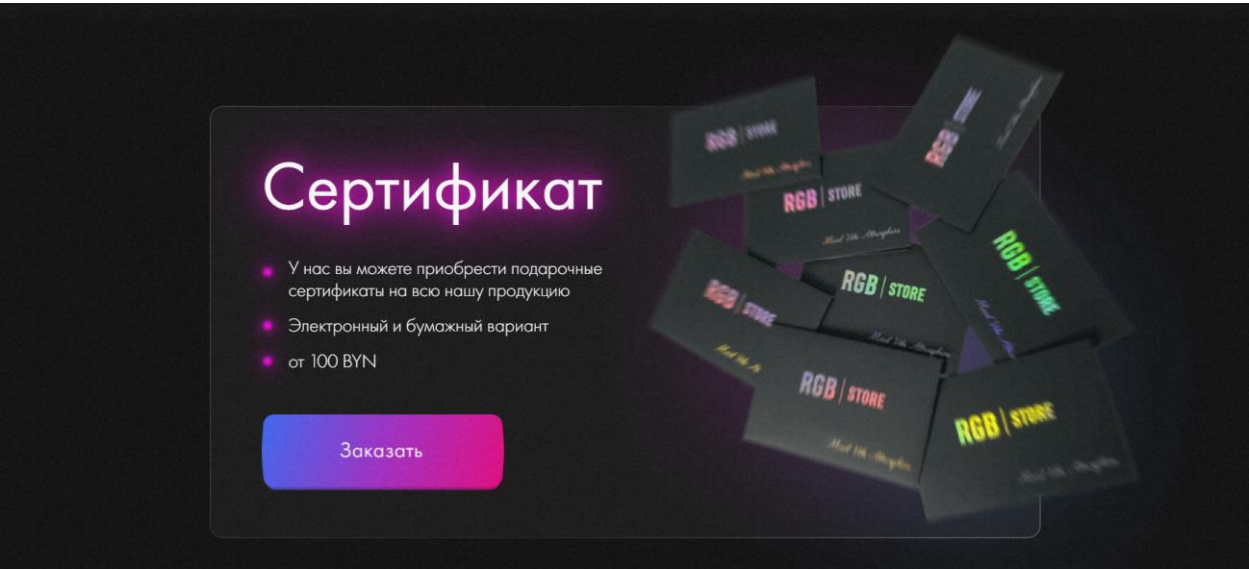


Рисунок 3 – блок сертификатов

Ниже располагается раздел сотрудничества (Рисунок 4). Его интересной особенностью является добавление голубого цвета кнопки в рамках логотипа мессенджера. Данная кнопка идеально вписывается в дизайн полученного раздела и делает некое разнообразие для него.

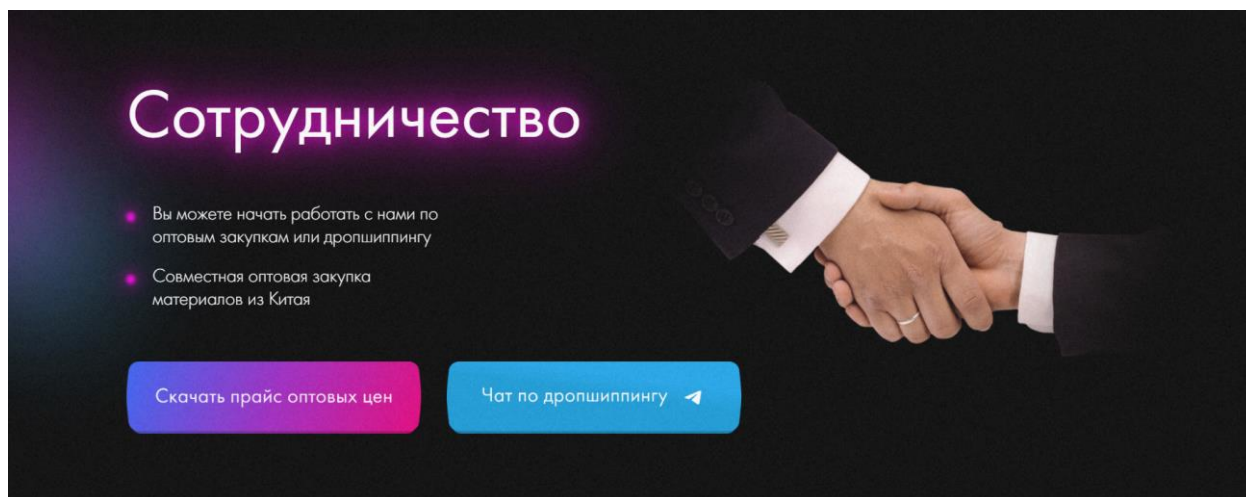


Рисунок 4 – раздел сотрудничества

Данная страница не является хорошим примером лендинга, однако при учете всех недостатков, замеченных в ходе анализа, можно реализовать хороший лендинг на основе данного. Из данного лендинга были взяты некоторые идеи по дизайну, а также некоторые элементы структуры.

1.4 Реализация дизайна лендинга

Основными цветами страницы стали фиолетовый и белый, практически в каждой её части можно найти их. Отличительной особенностью стало выделение фиолетовым цветом важных слов в предложениях и словосочетаниях.

Для сервиса было выбрано название «NeonWorld», поскольку это название отлично описывает то, что предлагает ассортимент онлайн-магазина.

При заходе на страницу, пользователь сразу видит перед собой логотип и преимущества выбранного сервиса. Немало важной особенностью является навигационное меню и контакты вверху страницы. Также имеется кнопка, побуждающая к покупке (Рисунок 5).

					СМА.509000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		8

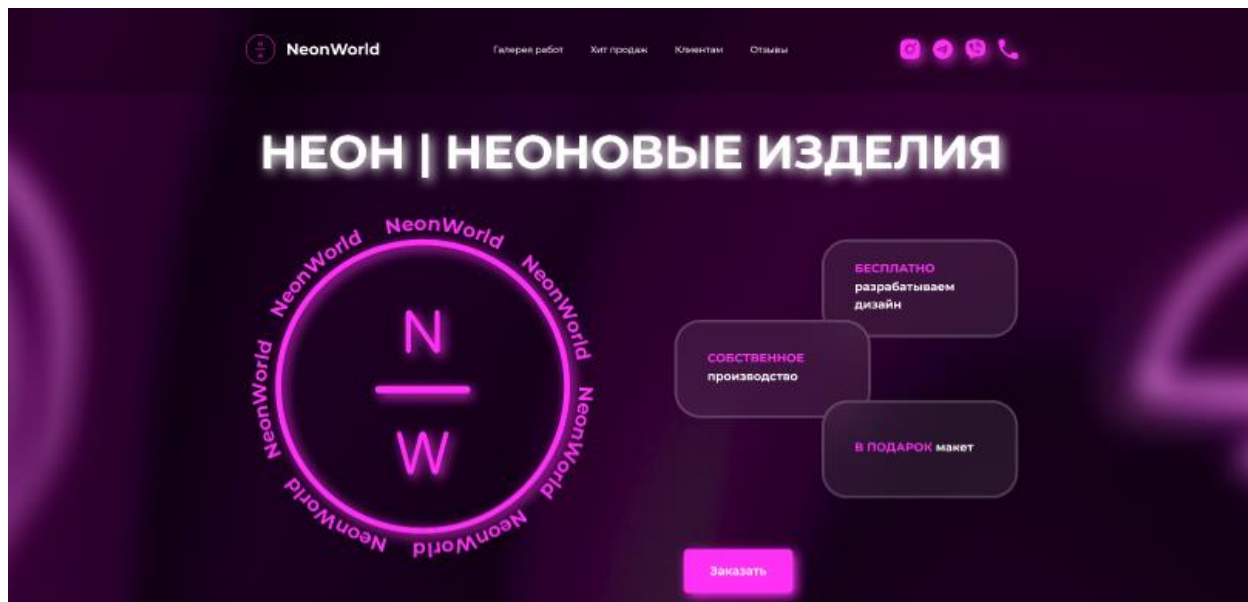


Рисунок 5 – верхняя часть страницы

Ниже можно найти числа из статистики магазина, которые дают понять зрителю, что из себя представляет данный сервис (Рисунок 6).

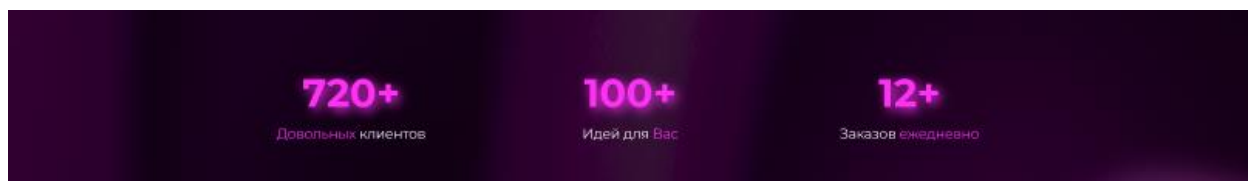


Рисунок 6 – числа для статистики

Далее перед посетителем появляется каталог неоновых вывесок, состоящий из фотографий реальных макетов (Рисунок 7).

					СМА.509000	Лист
						9
Изм.	Лист	№ докум.	Подпись	Дата		

НЕОНОВЫЕ ВЫВЕСКИ



Рисунок 7 – каталог неоновых вывесок

Еще чуть ниже располагается блок с каталогом, схожим с предыдущим, но уже предлагающий интерьерные панно (Рисунок 8).

ИНТЕРЬЕРНЫЕ ПАНННННО



Рисунок 8 – каталог интерьерных панно

Далее следует блок с ассортиментом продукции, именуемый «Хит Продаж» (Рисунок 9). Здесь представлены самые популярные товары нашей продукции, их описание, а также имеется кнопка, предполагающая подробное рассмотрение каждого изделия в качестве покупки.

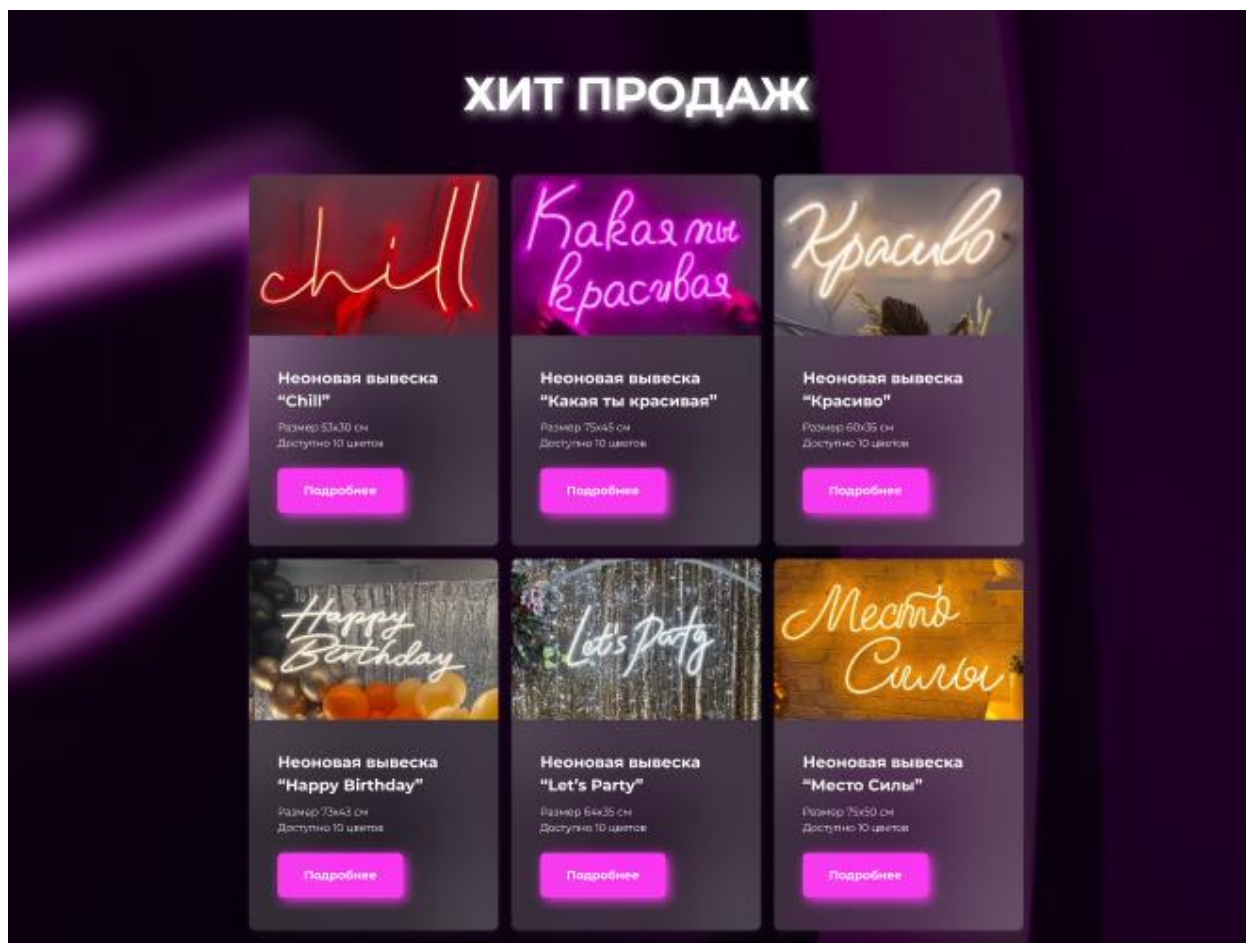


Рисунок 9 – блок с продукцией

Далее следует раздел, предоставляющий информацию об очередных преимуществах покупки в сервисе «NeonWorld» (Рисунок 10). Значки, использующиеся здесь, очень хорошо подходят к тексту и ещё раз убеждают покупателя в надежности сервиса.

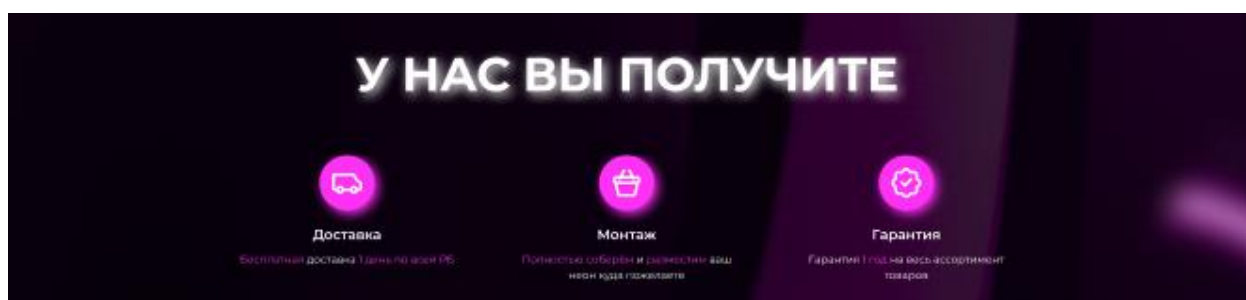


Рисунок 10 – раздел с преимуществами

Чуть ниже располагается раздел с отзывами (Рисунок 11). Отзывы оставлены людьми со всей Беларуси. В них описываются ощущения от покупки, эксплуатации и рассказывается о персонале сервиса.

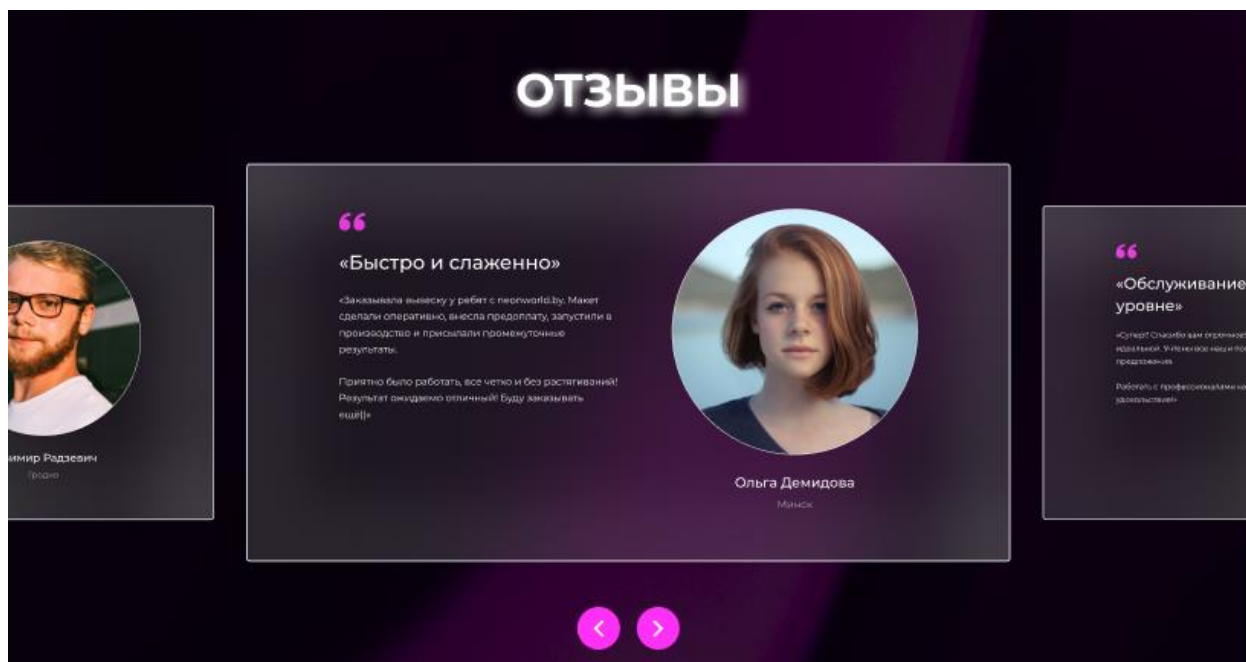


Рисунок 11 – раздел с отзывами

Завершается сайт разделом с дополнительной информацией и побуждающей кнопкой (Рисунок 12). Дополнительная информация представляет собой подвал с несколькими полезными ссылками.

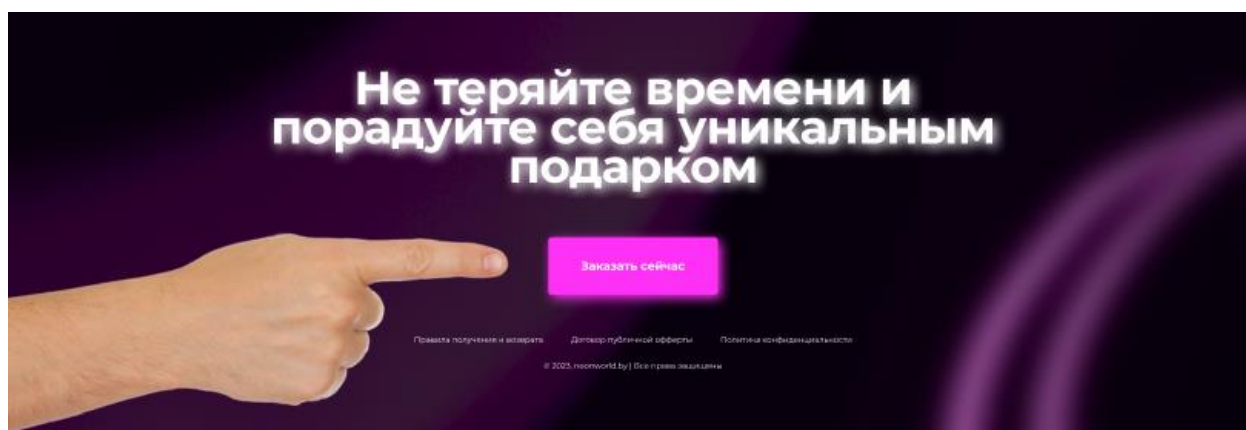


Рисунок 12 – раздел с дополнительной информацией

Стоит обратить внимание на то, что все блоки или разделы имеют единый стиль, ни капли не меняющийся на протяжении всей страницы. Дизайн очень яркий и запоминающийся. А также интерес к внешнему виду сайта добавляют неоновые элементы на заднем фоне, находящиеся в движении.

1.5 Вёрстка лендинга

Для верстки сайта были созданы три документа для работы с HTML, CSS, JavaScript neon.html, style.css и script.js соответственно. Для добавления всех необходимых ресурсов во время загрузки страницы, в заголовок были добавлены строки с подгрузкой шрифтов с внешнего сервера Google Fonts и

					СМА.509000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		13

загрузкой стилей и скриптов из локальных файлов. Также были установлены мета-параметры, которые позволяют повысить совместимость с различными видами устройств с разной шириной экрана (Листинг 1).

Листинг 1 – подгрузка ресурсов и установка мета-параметров

```
<head>
  <meta charset="UTF-8">
  <link rel='stylesheet' href='css/style.css'>
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com"
crossorigin>
  <link

href="https://fonts.googleapis.com/css2?family=Montserrat:wght@1
00;200;300;400;500;600;700;800;900&display=swap"
  rel="stylesheet">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>NeonWorld</title>
  <script src="script.js" defer></script>
</head>
```

В основной части страницы находятся все элементы, отображаемые впоследствии на экране. В основном это блоки <div>, содержащие в себе какое-либо наполнение, ссылки <a>, изображения и т.д. (Листинг 2).

Пример наполнения блока показан для контейнера с классом «header-container». Остальное содержимое блоков помечено комментариями для упрощения отображения структуры лендинга.

Листинг 2 – структура основной части лендинга

```
<body>
  <header>
    <div class="main-container">
      <div class="header-container">
        <div class="header-logo">
          <a href="#"> 
</a>
        </div>
        <div class='header-nav'>
          <a class='header-nav-item' href="#1">Галерея
работ</a>
          <a class='header-nav-item' href="#2">Хит
продаж</a>
          <a class='header-nav-item'
href="#3">Клиентам</a>
          <a class='header-nav-item'
href="#4">Отзывы</a>
        </div>
        <div class='header-contacts'>
          <div class='header-icon'>
```

```

        <a href="@">
            
        </a>
    </div>
    <div class='header-icon'>
        <a href="@">
            
        </a>
    </div>
    <div class='header-icon'>
        <a href="@">
            
        </a>
    </div>
    <div class='header-icon'>
        <a href="@">
            
        </a>
    </div>
</div>
</div>
</div>
</header>
<div class="main">
    <!--Блок с логотипом, панелями преимуществ и кнопкой-->
</div>
<div class="num">
    <!--Блок только с числами из статистики-->
</div>
<div class="num-description">
    <!--Блок с описанием чисел из статистики-->
</div>
<div id="1" class="vyveski">
    <!--Блок с названием каталога неоновых вывесок-->
</div>
<div class="gallery1">
    <!--Фото галереи неоновых вывесок-->
</div>
<div class="panno">
    <!--Блок с названием каталога панно-->
</div>
<div class="panno1">
    <!--Фото галереи интерьерных панно-->
</div>
<div id="2" class="hit">
    <!--Блок с названием раздела «Хит Продаж»-->
    <div class="hit-word">ХИТ ПРОДАЖ
    </div>
</div>
<div class="hit-container1">
    <!--3 карточки из раздела «Хит Продаж»-->

```



```

</div>
<div class="hit-container2">
    <!--3 карточки из раздела «Хит Продаж»-->
</div>
<div id="3" class="klientam">
    <!--Раздел «Клиентам» название-->
    <div class="klientam-word">
        У НАС ВЫ ПОЛУЧИТЕ
    </div>
</div>
<div class="klientam-container1">
    <!--Только иконки-->
</div>
<div class="klientam-container2">
    <!--Заголовок иконок-->
</div>
<div class="klientam-container3">
    <!--Строчки описания иконок-->
</div>
<div id="4" class="otzyvy">
    <!--Название раздела «ОТЗЫВЫ»-->
    <div class="otzyvy-word">
        ОТЗЫВЫ
    </div>
</div>
<div class="otzyvy-cards-container">
    <!--Контейнер карточек отзывов и стрелок навигации-->
</div>
<div class="end-container1">
    <!--Заголовок последнего раздела-->
    Не теряйте времени и<br>порадуйте себя
    уникальным<br>подарком
</div>
<div class="end-container2">
    <!--Подвал, фото руки и кнопка заказа-->
</div>
</body>

```

Для придания элементам страницы необходимого внешнего вида используется CSS. Он позволяет применить различные стили к объектам, например, расположить элемент, определить размер, цвет текста, изменить его шрифт, указать как именно содержимое контейнера будет в нём располагаться (Листинг 3).

Листинг 3 – реализация основных используемых стилей

```

.header-container {
    display: grid;
    grid-template: "a b c" auto / auto auto auto;
    justify-content: space-between;
    max-width: 1100px;
    width: 100%;

```

```

}

.header-logo {
    grid-area: a;
    display: flex;
}

.header-nav {
    grid-area: b;
    display: flex;
    padding-top: 12px;
    gap: 41px;
    justify-content: center;
}

.header-nav-item {
    display: flex;
    color: #fff;
    text-decoration: none;
    font-weight: 500;
    font-size: 14px;
    font-style: normal;
    line-height: 150%;

    transition: color 0.2s linear;
}

.header-nav-item:hover {
    color: #FF31F7;
}

.header-contacts {
    display: flex;
    padding-top: 6.5px;
    gap: 20px;
    grid-area: c;
    justify-content: end;
}

.header-icon {
    transition: drop-shadow 0.2s linear;
}

.header-icon:hover {
    filter: drop-shadow(5px 5px 20px #FF31F7);
}

```

Адаптация под размеры экрана реализована с помощью media-запросов. Они позволяют применять различные CSS-стили к объектам на основе каких-либо условий. В данном случае этим условием является ширина экрана. Таким образом, например, реализовано изменение размеров

					СМА.509000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		17

изображений, содержащихся в контейнере с классом «main-logo-img» (Листинг 4).

Листинг 4 – адаптация с использованием media-запросов

```
@media (max-width: 1500px) {  
    .main-logo-img {  
        height: 400px;  
        width: 400px;  
    }  
}  
  
@media (max-width: 400px) {  
    .main-logo-img {  
        height: 170px;  
        width: 170px;  
    }  
}
```

В CSS есть возможность создания псевдоклассов. Псевдоклассы позволяют управлять различными CSS-свойствами для различных состояний каких-либо объектов. В данном листинге используется два псевдокласса: «:hover», который применяет описанные CSS-стили только при наведении на объект, и «:active», который применяет стили при нажатии на объект. Например, при наведении на кнопку «Заказать» в начале страницы, она подсветится, а при нажатии на фото в галерее работ, оно размоется на короткий промежуток времени и вы перенаправитесь по ссылке (Листинг 5).

Листинг 5 – использование псевдоклассов

```
.button:hover {  
    box-shadow: 5px 5px 20px 0px #FF31F7;  
}  
.photo:active {  
    filter: blur(10px);  
}
```

Для реализации прокрутки карточек в горизонтальном направлении с помощью кнопок используется JavaScript. Сначала мы получаем ссылки на кнопки «Назад» и «Вперёд» из DOM-структуры и сохраняем их в переменные «prevButton» и «nextButton», а также получаем ссылки на все элементы с классом «card» и сохраняем их в переменной «cards». Затем мы создаем переменную «currentCardIndex» и устанавливаем ее равной индексу середины массива карточек. Это будет текущей отображаемой карточкой при инициализации. Функция «updateCards()» обновляет стили каждой карточки на основе их индекса относительно «currentCardIndex». Карточки, которые находятся слева или справа от текущей карточки, сдвигаются горизонтально (translateX) на определенное расстояние, а также масштабируются (scale) для создания эффекта глубины. Текущая карточка не масштабируется. Функция «updateButtons()» обновляет состояния кнопок «Назад» и «Вперёд». Если текущая карточка является первой, то кнопка «Назад» отключается, а если текущая карточка является последней, то кнопка «Вперёд» отключается.

После определения функций «updateCards()» и «updateButtons()», мы вызываем «updateCards()», чтобы инициализировать начальное состояние карточек. Далее, мы добавляем обработчики событий для кнопок "Вперёд" и «Назад». При нажатии на кнопку «Вперёд», «currentCardIndex» увеличивается на 1, но если текущая карточка уже последняя, то «currentCardIndex» не изменяется. Затем вызывается «updateCards()» и «updateButtons()» для обновления отображаемых карточек и состояния кнопок. Аналогично, при нажатии на кнопку «Назад», «currentCardIndex» уменьшается на 1, но если текущая карточка уже первая, то «currentCardIndex» не изменяется. Затем также вызываются «updateCards()» и «updateButtons()». В итоге, этот код создает интерактивный компонент, который позволяет пользователю прокручивать набор карточек горизонтально с помощью кнопок «Назад» и «Вперёд», обеспечивая эффект переключения между карточками с анимацией (Листинг 6).

Листинг 6 – реализация прокрутки карточек в горизонтальном направлении

```
const prevButton = document.querySelector('#prevButton');
const nextButton = document.querySelector('#nextButton');

const cards = document.querySelectorAll('.card');
let currentCardIndex = Math.floor(cards.length / 2);

function updateCards() {
  for (const [index, card] of Object.entries(cards)) {
    let i = parseInt(index);

    const delta = i - currentCardIndex;

    // translateX(100%) scale(0.9)
    card.style.transform = `translateX(${delta}00%)` + (i !==
currentCardIndex ? " scale(0.8)" : "");
  }
}

function updateButtons() {
  prevButton.disabled = currentCardIndex == 0;
  nextButton.disabled = currentCardIndex == cards.length - 1;
}

updateCards();

nextButton.addEventListener("click", () => {
  currentCardIndex++;
  if (currentCardIndex > cards.length - 1)
    currentCardIndex--;

  updateCards();
  updateButtons();
});
```

```

prevButton.addEventListener("click", () => {
    currentCardIndex--;
    if (currentCardIndex < 0)
        currentCardIndex++;

    updateCards();
    updateButtons();
});

```

					СМА.509000	Лист
						20
Изм.	Лист	№ докум.	Подпись	Дата		

ЗАКЛЮЧЕНИЕ

Практика по созданию лендинга оказалась важной для понимания современных тенденций в мире веб-разработки и маркетинга.

Создание дизайна и верстка лендингов – это искусство сочетания креативности и технической компетенции. Лендинги играют важную роль в привлечении и удержании клиентов, продвижении товаров и услуг, а также в сборе данных о целевой аудитории, и умение создавать их эффективно становится всё более ценным навыком. Эта практика не только позволила понять суть создания лендингов, но и подготовила к решению задач в области веб-разработки и дизайна в будущем.

В ходе прохождения практики были получены как практические, так и теоретические навыки. К ним можно отнести навыки работы с различными средствами, такими как Figma и Visual Studio Code, навыки веб-разработки и дизайна, включая HTML, CSS и JavaScript.

					СМА.509000	Лист
Изм.	Лист	№ докум.	Подпись	Дата		21

СПИСОК ИСТОЧНИКОВ

1 Что такое лендинг, наш ТОП-60 лучших лендингов [Электронный ресурс]. – Режим доступа: <https://www.uplab.ru/blog/50-ubeditelnykh-russkoyazychnykh-lendingov-1/>. – Дата доступа: 28.06.2023.

2 Figma - описание платформы, важные особенности [Электронный ресурс]. – Режим доступа: <https://liquidhub.ru/blogs/blog/figma>. – Дата доступа: 28.06.2023.

3 Редактор кода Visual Studio Code: подробный гайд по настройке и установке плагинов [Электронный ресурс]. – Режим доступа: <https://timeweb.com/ru/community/articles/cto-takoe-visual-studio-code>. – Дата доступа: 28.06.2023.