МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование темы проекта или работы)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к отчету по лабораторной работе

по дисциплине

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование дисциплины)

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Савкин А.Е. .

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_Гарасев М.Г. .

(подпись) (фамилия, и.,о.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(шифр группы)

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Задание к выполнению лабораторной работы № 2**

**Создание веб-сайтов**

**Задание на лабораторную работу:**

Разработать веб-сайт с использованием любых доступных инструментов — от HTML/CSS/JS до фреймворков или конструкторов, с условием наличия не менее трёх содержательных вкладок, стилизованных элементов и, при необходимости, интерактивных скриптов.

**Цель работы:**

Создание креативного веб-сайта с использованием современных инструментов веб-разработки, освоение базовых принципов вёрстки, применения CSS и JavaScript.

**Ход работы:**

1. Выбрана ниша для разработки веб-сайта – ремонт техники Apple.

2. В качестве наиболее подходящего инструмента для реализации проекта была выбрана платформа Tilda Publishing.

3. Разработана структура будущего веб-сайта:

3.1. Шапка веб-сайта для быстрой навигации

3.2. Главная страница с названием сервиса и соответствующим фото

3.3. Описание компании

3.4. Описание услуг (детализация)

3.4. Блок с отзывами о работе

3.5. Описание команды (сотрудников)

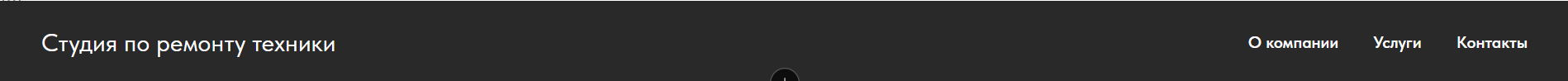
3.6. Перечень предоставляемых услуг и цен

3.7. Контактная информация

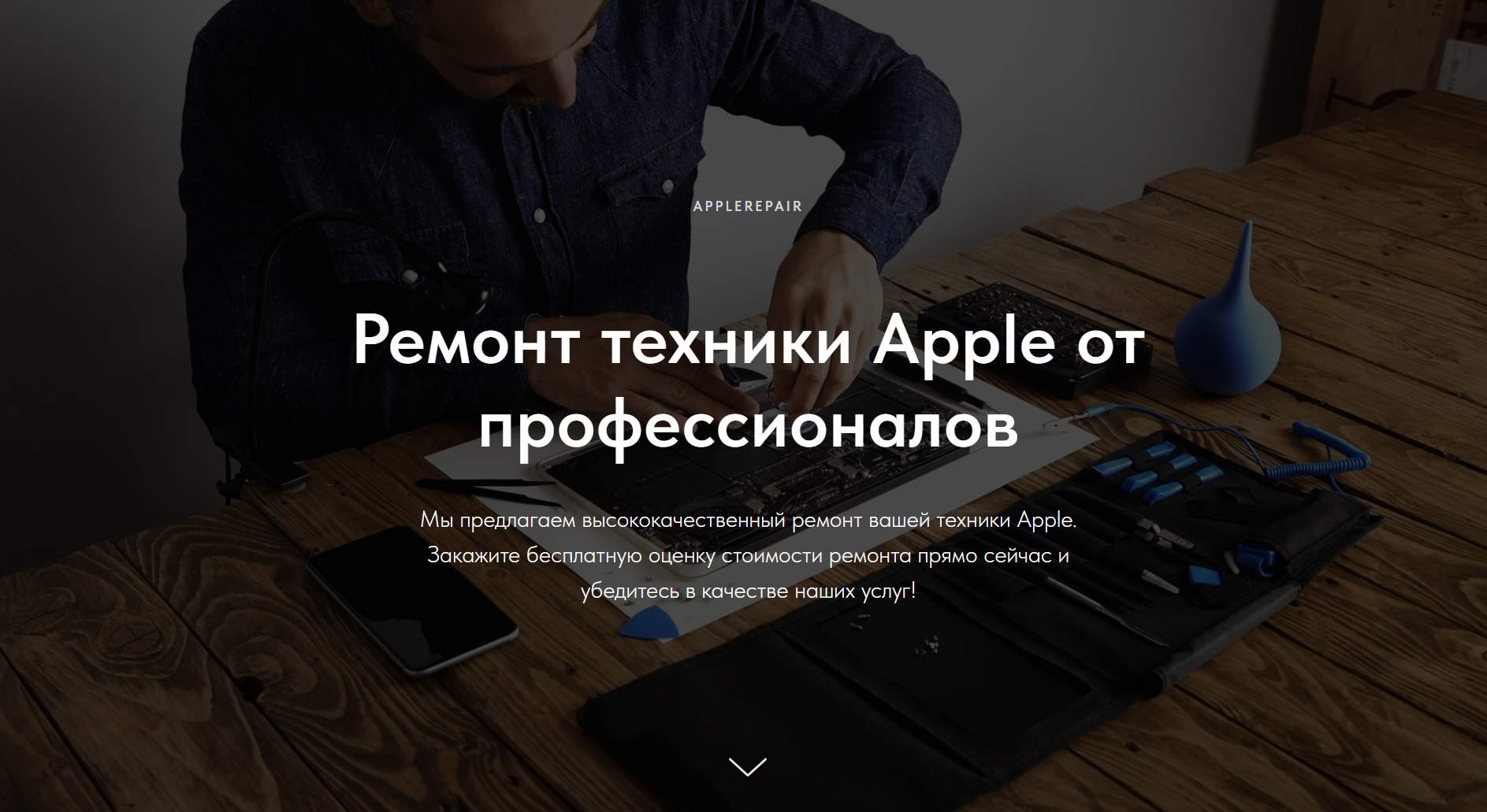
3.8. Форма для сбора обратной связи.

4. Созданы отдельные блоки веб-сайта в соответствие с ранее разработанной структурой, наполнили страницы контентом с помощью встроенного в платформу Tilda Publishing искусственного интеллекта:

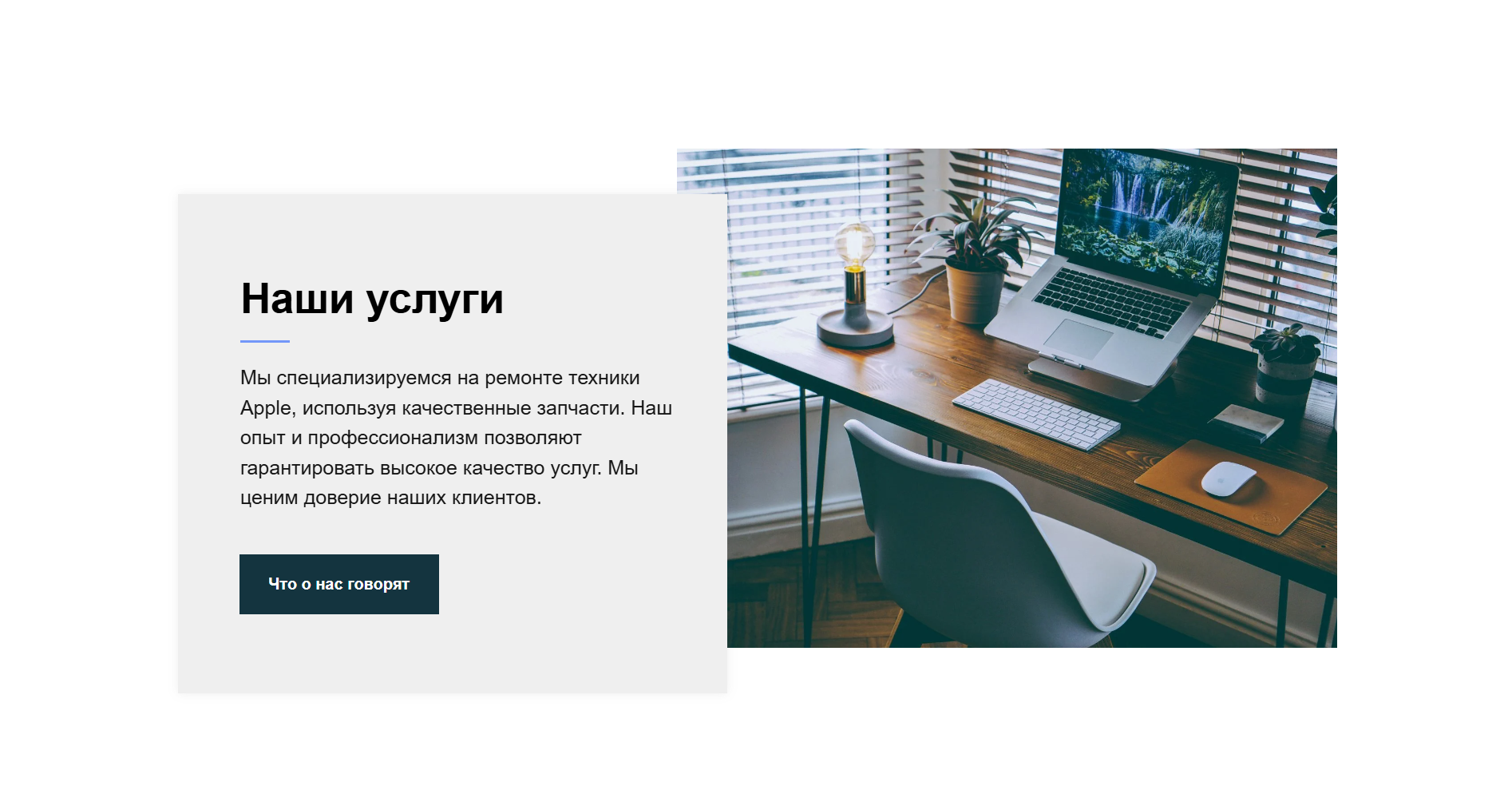
4.1. Шапка веб-сайта с кликабельными элементами:



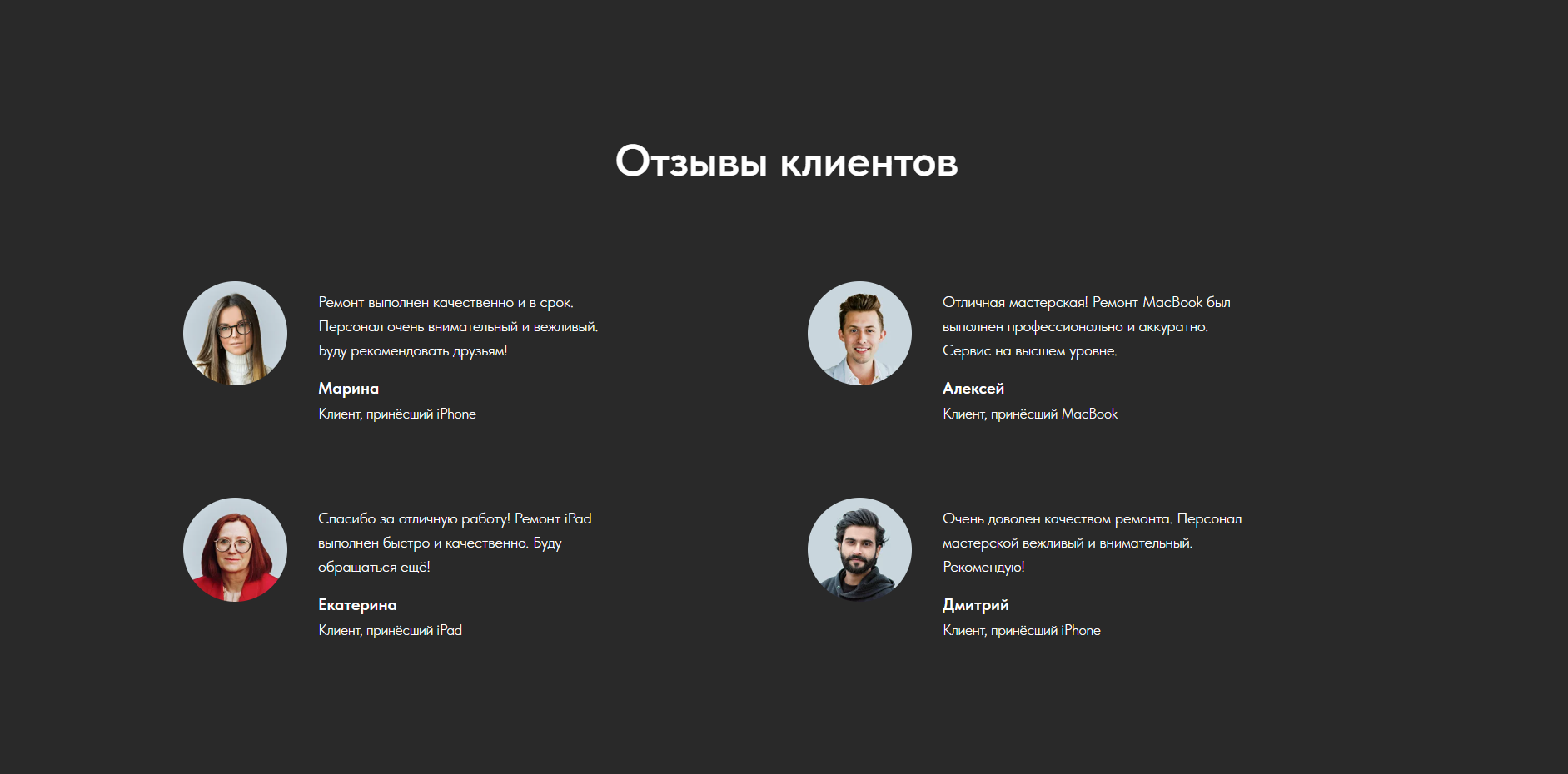
4.2. Главная страница с названием сервиса и кратким описанием:



4.3. Описание компании и спектра предоставляемых услуг в общем виде:



4.3.1. Добавлена кнопка «что о нас говорят» для перехода на вкладку с отзывами клиентов:



4.3.2. На втором блоке данной вкладки была применена анимация кнопки при наведении с помощью CSS-кода:

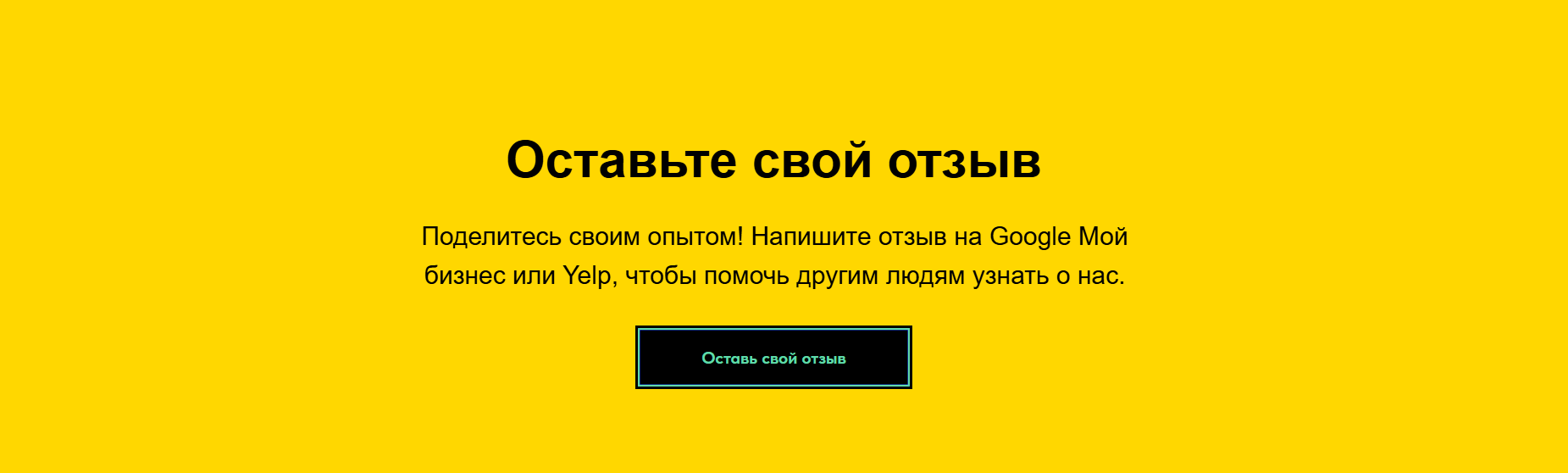
Изображение выглядит как текст, снимок экрана, Шрифт, желтый

Содержимое, созданное искусственным интеллектом, может быть неверным.

4.3.2.1. Добавлен CSS-код в специальный блок платформы Tilda (сам код и его описание представлены в Приложении 1):

Изображение выглядит как текст, снимок экрана, программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

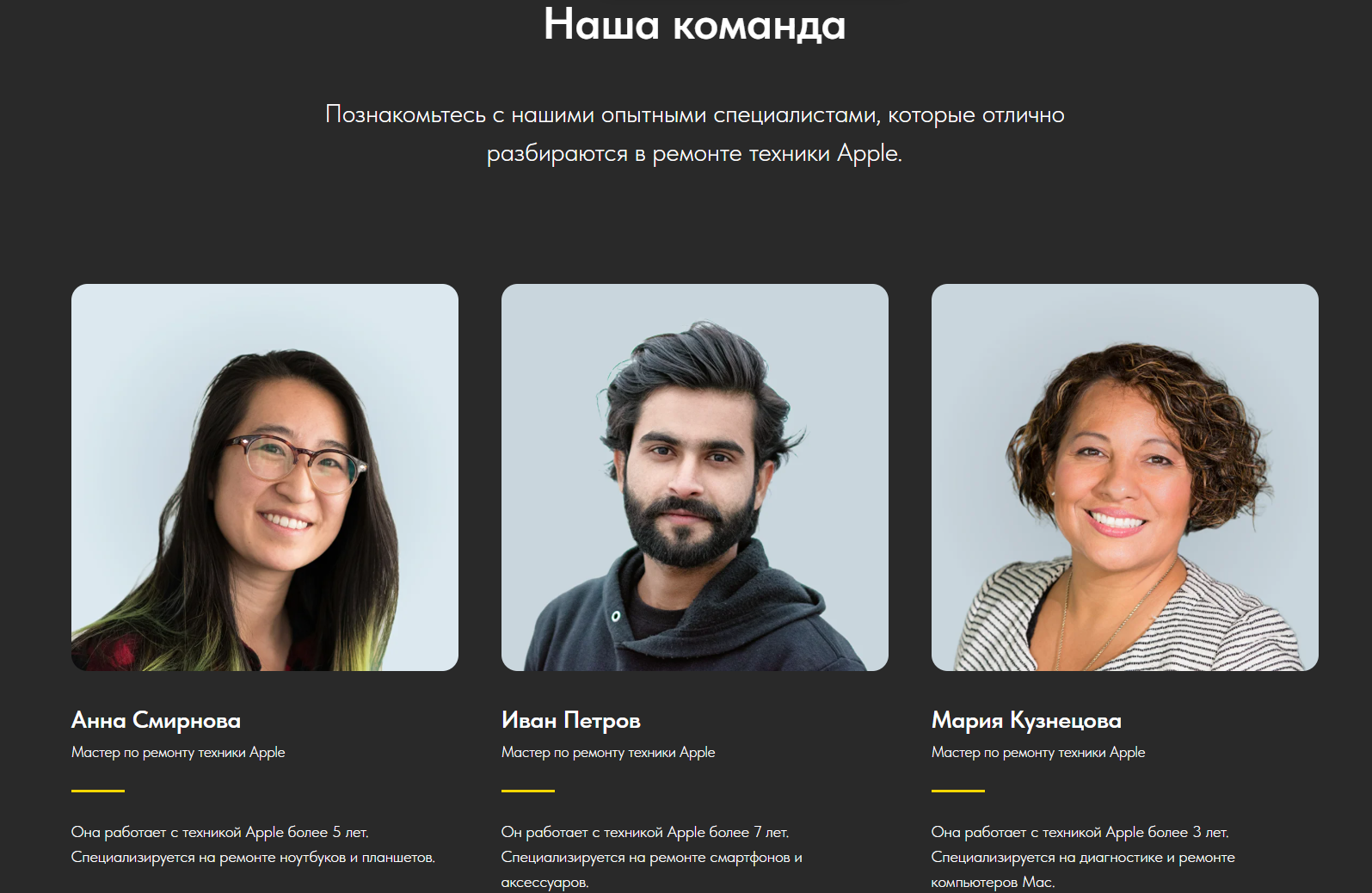


4.3.2. На втором блоке данной вкладки располагается кнопка, переводящая на вкладку с формой для сбора обратной связи:

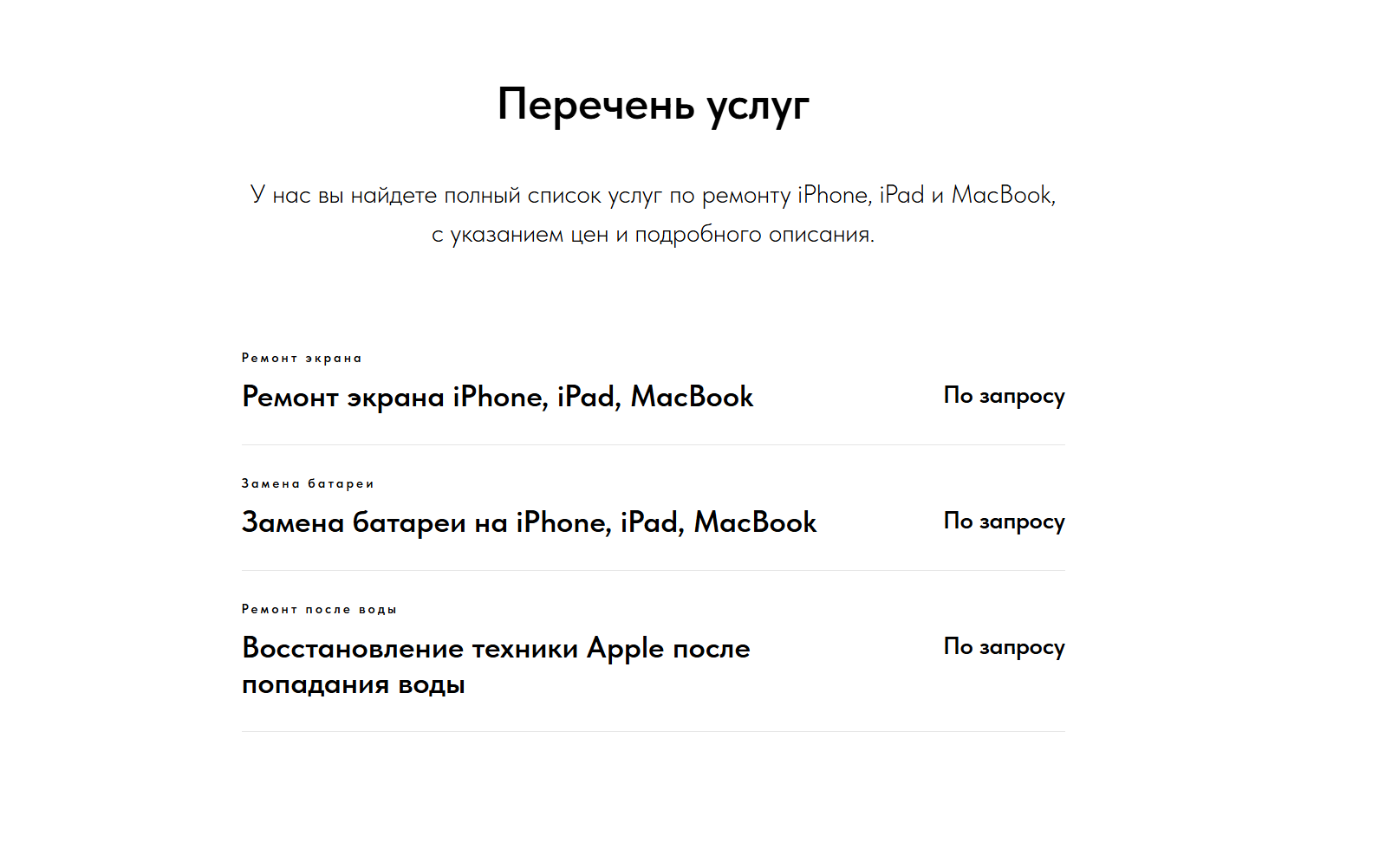
Изображение выглядит как текст, снимок экрана, Шрифт, желтый

Содержимое, созданное искусственным интеллектом, может быть неверным.

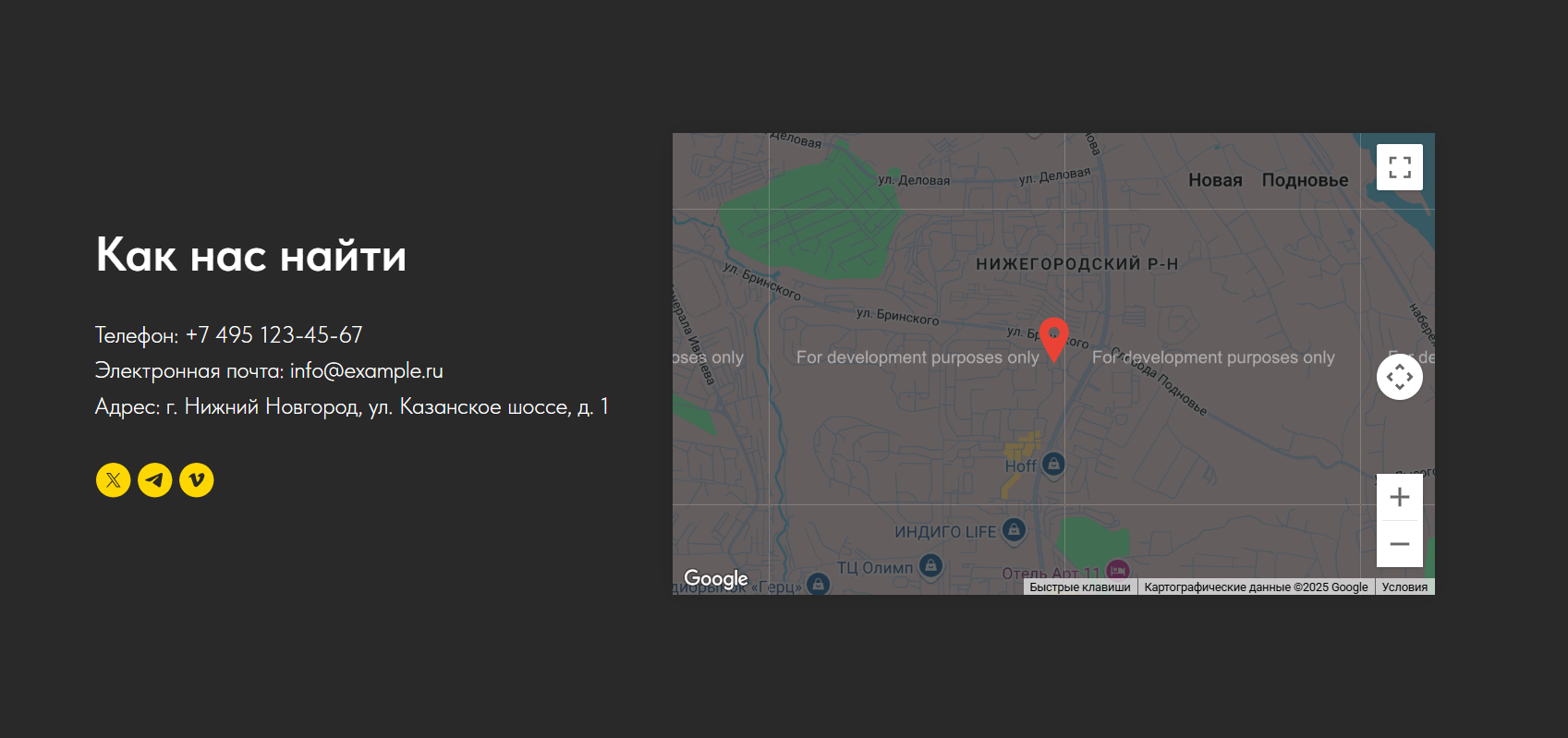
4.4. Описание команды с фото:



4.5. Описание услуг (детализация):



4.6. Контактная информация (дополнительно добавлен виджет карты для удобства):



5. Добавлена микроразметка для улучшения SEO с помощью HTML тегов. Микроразметка помогает поисковым системам лучше понимать контент.

5.1. Добавлен HTML тег в специальный блок платформы Tilda (сам код и его описание представлены в Приложении 2):

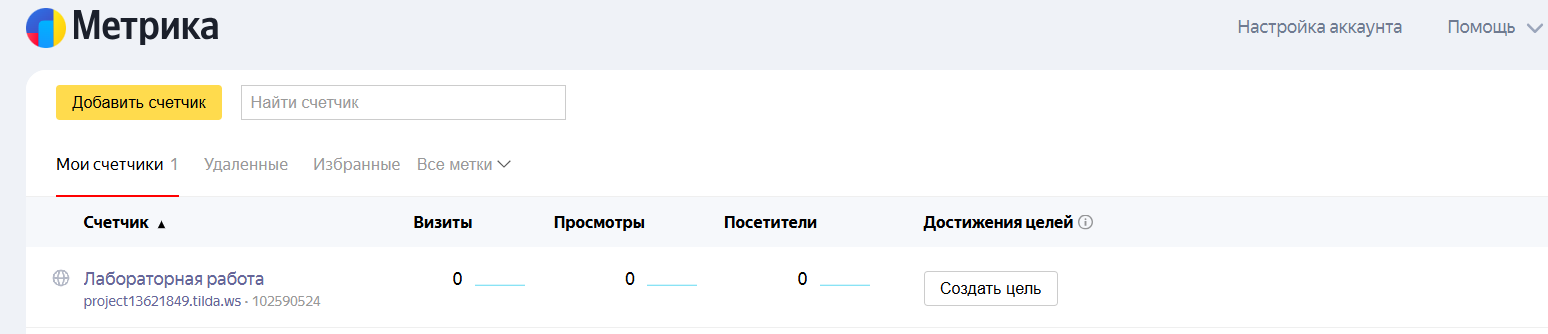


6. Применена отправка событий в Яндекс.Метрику о достижении цели при клике на ссылку/кнопку с помощью скрипта JS.

6.1. Добавлен скрипта JS в специальный блок платформы Tilda (сам код и его описание представлены в Приложении 3):



6.2. В личном кабинете Яндекс.Метрики был создан счетчик, к номеру которого и привязан скрипт.



**Ссылка на сайт:** [**http://project13621849.tilda.ws**](http://project13621849.tilda.ws/)

**Вывод:**

В ходе лабораторной работы был разработан тематический веб-сайт сервиса по ремонту техники Apple на платформе Tilda Publishing. Были освоены базовые принципы веб-разработки: создание структуры сайта, стилизация с помощью CSS, добавление анимации, работа с микроразметкой для SEO и применение JavaScript для отслеживания пользовательских действий. Получены практические навыки работы с визуальным редактором, базовыми веб-технологиями и аналитическими инструментами.

Таким образом, достигнута цель работы — создание креативного, функционального сайта с использованием современных веб-инструментов.

**Приложение**

**Приложение 1**

**Применение CSS**

Анимация при наведении на кнопки

<style>

.uc-features .t-btn {

box-shadow: inset 0 0 0 2px #1f5bff;

position: relative;

transition: color 0.25s;

box-sizing: border-box;

}

.uc-features .t-btn::before,

.uc-features .t-btn::after {

content: '';

position: absolute;

border: 2px solid transparent;

width: 0;

height: 0;

top: 0;

left: 0;

box-sizing: inherit;

}

.uc-features .t-btn:hover {

color: #60daaa !important;

}

.uc-features .t-btn:hover::before,

.uc-features .t-btn:hover::after {

width: 100%;

height: 100%;

}

.uc-features .t-btn:hover::before {

border-top-color: #60daaa;

border-right-color: #60daaa;

transition: width 0.15s ease-out, height 0.15s ease-out 0.15s;

}

.uc-features .t-btn:hover::after {

border-bottom-color: #60daaa;

border-left-color: #60daaa;

transition: height 0.15s ease-out, width 0.15s ease-out 0.15s;

}

</style>

**Описание кода:**

Стилизация кнопки .t-btn, находящейся внутри блока .uc-features. При наведении на кнопку появляется анимированная рамка по краям, а цвет текста меняется

Изначально в настройка кнопки BLOCK CSS CLASS NAME был добавлен атрибут «features»

Далее добавлен HTML-код в специальный блок:

**Блок 1: Основной стиль кнопки**

.uc-features .t-btn {

box-shadow: inset 0 0 0 2px #1f5bff;

position: relative;

transition: color 0.25s;

box-sizing: border-box;

}

* **.t-btn внутри .uc-features** — это целевая кнопка.
* box-shadow: inset — создаёт внутреннюю синюю рамку.
* position: relative — позволяет точно позиционировать псевдоэлементы ::before и ::after.
* transition: color 0.25s — плавное изменение цвета текста при наведении.
* box-sizing: border-box — включает отступы и границы в общую ширину/высоту элемента.

**Блок 2: Подготовка псевдоэлементов**

.uc-features .t-btn::before,

.uc-features .t-btn::after {

content: '';

position: absolute;

border: 2px solid transparent;

width: 0;

height: 0;

top: 0;

left: 0;

box-sizing: inherit;

}

* Создаёт два пустых псевдоэлемента (до и после) поверх кнопки.
* Устанавливается абсолютно в верхний левый угол (top: 0, left: 0).
* Изначально границы прозрачные, размеры — 0 (невидимые).

**Блок 3: Эффект при наведении**

.uc-features .t-btn:hover {

color: #60daaa !important;

}

* При наведении цвет текста кнопки меняется на зелёный (#60daaa).

**Блок 4: Анимация рамки при наведении**

.uc-features .t-btn:hover::before,

.uc-features .t-btn:hover::after {

width: 100%;

height: 100%;

}

* При наведении ::before и ::after расширяются на весь размер кнопки.

**Блок 5: Цвет и анимация границ ::before**

.uc-features .t-btn:hover::before {

border-top-color: #60daaa;

border-right-color: #60daaa;

transition: width 0.15s ease-out, height 0.15s ease-out 0.15s;

}

* Появляется **верхняя и правая** рамка (анимация по ширине и затем по высоте).

**Блок 6: Цвет и анимация границ ::after**

.uc-features .t-btn:hover::after {

border-bottom-color: #60daaa;

border-left-color: #60daaa;

transition: height 0.15s ease-out, width 0.15s ease-out 0.15s;

}

* Появляется **нижняя и левая** рамка с противоположной задержкой (сначала высота, затем ширина).

**Итог:**

При наведении на кнопку создаётся эффект "прорастающей" рамки с четырёх сторон, зелёного цвета, с плавной анимацией. Этот приём улучшает интерактивность и визуальную привлекательность кнопок.

**Приложение 2**

**Использование HTML тегов**

Добавление микроразметки (Schema.org) для улучшения SEO

<script type="application/ld+json">

{

"@context": "https://schema.org",

"@type": "LocalBusiness",

"name": "AppleRepair",

"telephone": "+7 (495) 123-45-67",

"address": {

"@type": "PostalAddress",

"streetAddress": "ул. Казанское шоссе, 1",

"addressLocality": "Нижний Новгород",

"addressCountry": "RU"

},

"openingHoursSpecification": {

"@type": "OpeningHoursSpecification",

"dayOfWeek": ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday"],

"opens": "10:00",

"closes": "20:00"

},

"priceRange": "₽₽"

}

</script>

**Описание кода:**

Этот блок — это **структурированные данные в формате JSON-LD**, встроенные в HTML через тег <script type="application/ld+json">. Он описывает **организацию типа "LocalBusiness"** — локальный бизнес "AppleRepair" со сведениями, понятными для поисковых систем (Google, Яндекс и др.).

**Описание элементов:**

<script type="application/ld+json">

* Указывает, что внутри находится скрипт с **структурированной информацией** в формате JSON-LD.

"@context": "https://schema.org"

* Контекст — ссылка на словарь схемы **Schema.org**, используемой для описания структурированных данных.

"@type": "LocalBusiness"

* Тип объекта — **локальный бизнес**, подходящий для сервисов, магазинов, кафе и т.п.

"name": "AppleRepair"

* Название организации — отображается в сниппетах или справочниках.

"telephone": "+7 (495) 123-45-67"

* Контактный номер телефона.

"address": {

"@type": "PostalAddress",

...

}

* Вложенный объект описывает **почтовый адрес** организации:
  + streetAddress: улица и номер здания;
  + addressLocality: город;
  + addressCountry: страна.

"openingHoursSpecification": {

...

}

* График работы:
  + dayOfWeek: дни недели;
  + opens и closes: время открытия и закрытия.

"priceRange": "₽₽"

* Условный диапазон цен (например, ₽ – дешево, ₽₽ – средне, ₽₽₽ – дорого). Это поле может отображаться в результатах поиска.

**Зачем это нужно:**

Такая микроразметка помогает:

* Улучшить **SEO** сайта;
* Отображать бизнес в **расширенных сниппетах** в поисковых системах;
* Повысить **понятность информации для поисковых ботов**.

**Итог:**

Это современный способ добавить **структурированные данные** на сайт, повышая видимость и достоверность информации в поиске. Код не отображается для пользователя напрямую, но помогает поисковым системам правильно интерпретировать содержание страницы.

**Приложение 2**

**Применение скриптов JS**

Отправка события в Яндекс.Метрику о достижении цели при клике на ссылку/кнопку

<script>

if (document.readyState !== 'loading') {

us\_sendMertics();

} else {

document.addEventListener('DOMContentLoaded', us\_sendMertics);

}

function us\_sendMertics() {

var links = document.querySelectorAll('[href="любая\_ссылка"]');

Array.prototype.forEach.call(links, function (link) {

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

if (window.mainMetrikaId !== undefined) {

yaCounter102590524.reachGoal('произвольная\_цель');

}

});

});

}

</script>

**Описание кода:**

**Что делает данный скрипт:**

Скрипт отслеживает **нажатие на конкретную ссылку** на сайте и, при клике, **отправляет событие (цель)** в Яндекс.Метрику. Это используется для анализа пользовательского поведения.

if (document.readyState !== 'loading') {

us\_sendMertics();

} else {

document.addEventListener('DOMContentLoaded', us\_sendMertics);

}

* Проверяется, загружен ли документ:
  + Если уже загружен — вызывается функция us\_sendMertics() сразу.
  + Если ещё нет — она будет вызвана при наступлении события DOMContentLoaded.

function us\_sendMertics() {

var links = document.querySelectorAll('[href="любая\_ссылка"]');

* Выбираются **все ссылки на странице с заданным атрибутом href="любая\_ссылка"** (его нужно заменить на реальный адрес, по которому осуществляется переход, например: href="#order").

Array.prototype.forEach.call(links, function (link) {

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

* Для каждой найденной ссылки добавляется **обработчик события click**.

if (window.mainMetrikaId !== undefined) {

yaCounter102590524.reachGoal('произвольная\_цель');

}

* Если метрика активна (mainMetrikaId определена), вызывается функция reachGoal — **отправка пользовательского действия (цели)** в конкретный счетчик Яндекс.Метрики (yaCounter102590524).
* 'произвольная\_цель' — это идентификатор цели, заданный в интерфейсе Яндекс.Метрики (например: 'order\_click', 'form\_submit' и др.).

**Назначение:**

* **Отслеживание конверсий**: позволяет понять, сколько пользователей нажимают на ключевые кнопки/ссылки.
* **Аналитика эффективности элементов** сайта.
* **Настройка ретаргетинга** или рекламных кампаний на основе действий пользователя.