Государственное автономное учреждение

Калининградской области

профессиональная образовательная организация

«Колледж предпринимательства»

**Курсовая работа**

**Тема: «Разработка лэндинга "Ведра"»**

Выполнил:

обучающийся гр. ИСП 22-22 специальность 09.02.07  
Информационные системы и программирование

Я.А. Кононов

руководитель

М.В. Зверев

Калининград

2025

# 

# **СОДЕРЖАНИЕ**

3 см.

[ВВЕДЕНИЕ 4](#_Toc515549762)

[1. ЛЕНДИНГ 6](#_Toc515549763)

[1.1. **Блок “Шапка сайта”** 6](#_Toc515549764)

[1.2 **Блок “**Баннер**”** 7](#_Toc515549765)

[1.3 **Блок “**О продукте**”** 9](#_Toc515549766)

[1.4 **Блок “**каталог**”** 12](#_Toc515549767)

[1.5 **Блок “К**оментарии**”** 13](#_Toc515549769)

[1.6 **Блок “Обратная связь”** 14](#_Toc515549769)

[1.7 **Блок “**Подвал сайта**”** 18](#_Toc515549769)

[**ЗАКЛЮЧЕНИЕ 19**](#_Toc515549777)

СПИСОК [ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 21](#_Toc515549778)

**СПИСОК ТЕРМИНОВ И СОКРАШЕНИЙ**

HTML - Язык гипертекстовой разметки

CSS - Каскадные таблицы стилей

JS - JavaScript

JavaScript - это **высокоуровневый, интерпретируемый, мультипарадигменный язык программирования**, используемый в основном для создания интерактивных веб-страниц.

**Express.js — фреймворк для создания веб-приложений на языке JavaScript**

**Node.js — это бесплатная среда с открытым исходным кодом, предназначенная для выполнения кода JavaScript на сервере.**

# **ВВЕДЕНИЕ**

Данная курсовая работа посвящена разработке лендинга (посадочной страницы) для необычного товара — ведра. Выбор этого предмета обусловлен желанием продемонстрировать, что даже для, казалось бы, простого и обыденного продукта можно создать эффективный и привлекательный веб-ресурс, способный повысить продажи. Актуальность темы подтверждается растущей потребностью в качественном цифровом маркетинге, где лендинги играют ключевую роль в привлечении целевой аудитории и увеличении конверсии. В рамках данной работы лендинг будет дополнен функционалом серверной части, обеспечивающим взаимодействие с пользователем и базой данных.

**Цель работы** — разработка функционального и эстетичного лендинга, включающего серверную часть, который эффективно представляет ведро как товар и мотивирует пользователей к покупке. Для достижения этой цели необходимо решить следующие **задачи**:

1. **Анализ целевой аудитории:** определить основные характеристики потенциальных покупателей ведер (частные лица, строительные компании, фермерские хозяйства и т.д.) и их потребности.
2. **Разработка концепции лендинга:** определите основную идею и стиль лендинга, учитывая целевую аудиторию и особенности товара.
3. **Дизайн и верстка:** создать визуально привлекательный и удобный для восприятия дизайн лендинга, обеспечивающий интуитивно понятную навигацию, используя фреймворк Tailwind CSS.
4. **Реализация функциональности:** внедрить необходимые элементы интерактивности, анимацию, для повышения вовлеченности пользователей, включая генерацию QR-кодов.
5. **Разработка серверной части:** создание серверной части на Node.js с использованием фреймворка Express.js для обработки запросов, взаимодействия с базой данных и отправки электронных писем.
6. **Взаимодействие с базой данных:** реализация взаимодействия с базой данных SQLite3 для хранения информации о заказах и других данных.
7. **Адаптивная вёрстка:** обеспечить корректное отображение лендинга на различных устройствах (десктопах, планшетах, смартфонах).
8. **Отправка электронных писем:** использование библиотеки Nodemailer для отправки уведомлений о новых заказах.

**Целевая аудитория** лендинга достаточно широка и включает в себя как частных лиц (для бытовых нужд), так и представителей бизнеса (строительные, сельскохозяйственные компании и др.). **Основная идея** лендинга будет заключаться в демонстрации функциональных характеристик ведра (прочность, объем, удобство использования).

**Основные технологии**, используемые в данной работе: HTML5, CSS3, JavaScript, **Tailwind CSS**, **Node.js**, **Express.js**, **SQLite3**, **Nodemailer**, **path**, **qrcode**, а также инструменты для создания адаптивной вёрстки и SEO-оптимизации. Результатом работы станет готовый лендинг с функциональной серверной частью, демонстрирующий эффективность применения современных веб-технологий для продвижения даже самых, казалось бы, обычных товаров.

# **ЛЕНДИНГ**

Лендинг — это веб-страница, задача которой — побудить пользователя совершить целевое действие. Например, сделать покупку, подписаться на рассылку, скачать бесплатную книгу или чек-лист.

**1.1. Блок “Шапка сайта”**

Шапка сайта — важный элемент целевой страницы, поскольку она формирует первое впечатление у пользователя. В данном проекте шапка сайта включает в себя логотип онлайн-курса, навигационное меню (ссылки на разделы страницы),логотип и анимированный фон с пузырьками.

**Технологии:** **HTML** для структуры, **CSS** для стилизации,адаптивности дизайна и эффектов наведения на ссылки, **JavaScript** для анимации пузырей, **qrcode для получения ссылки на Гитхаб**.

**Анимационные эффекты:**

* **Анимация пузырей (JavaScript):** на фоне шапки сайта реализована анимация, создающая эффект плавно поднимающихся пузырей. Это добавляет динамичности и привлекательности, не отвлекая от основной информации. Пузыри имеют разные размеры и скорость подъема, что создает более реалистичную картину. Анимация реализована с использованием JavaScript, что обеспечивает высокую производительность и плавность.
* **Эффект наведения на ссылки (CSS и JavaScript):** при наведении курсора на пункты меню (ссылки) происходит появление 2 теней у текста что образует эффект дублирования текста.

**Обоснование:** Анимация пузырей была выбрана для создания легкой и непринужденной атмосферы, ассоциирующейся с обучением и развитием. Эффект наведения на ссылки улучшает пользовательский опыт, делая навигацию более интуитивной и удобной.

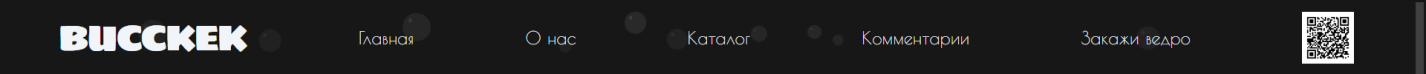
****

Рисунок 1. Хедер на Персональном компьютере

****

Рисунок 2. Хедер на телефоне

**1.2. Блок “Баннер”**

Баннер – это центральный визуальный элемент Landing Page, призванный привлечь внимание пользователя и кратко рассказать о предлагаемом онлайн-курсе. В данном проекте баннер содержит анимацию, отражающее тематику веб-дизайна, заголовок, таймер.

**Технологии:** HTML, CSS, JavaScript для анимации.

**Анимационные эффекты:**

* **Эффект «глюка» в заголовке (CSS):** заголовок курса подвергается визуальному искажению, имитирующему эффект «глюка» или помех. Этот эффект достигается с помощью CSS-анимации и создания нескольких слоев текста, смещенных относительно друг друга и меняющих цвет. Эффект добавляет динамичности и делает заголовок более запоминающимся.
* **Таймер до окончания акции (JavaScript):** отображает обратный отсчет времени до завершения специального предложения.Таймер стимулирует пользователя принять решение о покупке.
* **Интерактивное ведро (JavaScript и CSS):** пользователь может «наполнять» ведро, нажимая на кнопки добавления камней, песка и воды. При добавлении каждого элемента ведро анимированно «наполняется».Так же пользователь может убрать всё что есть в ведре нажав на кнопку сбросить.

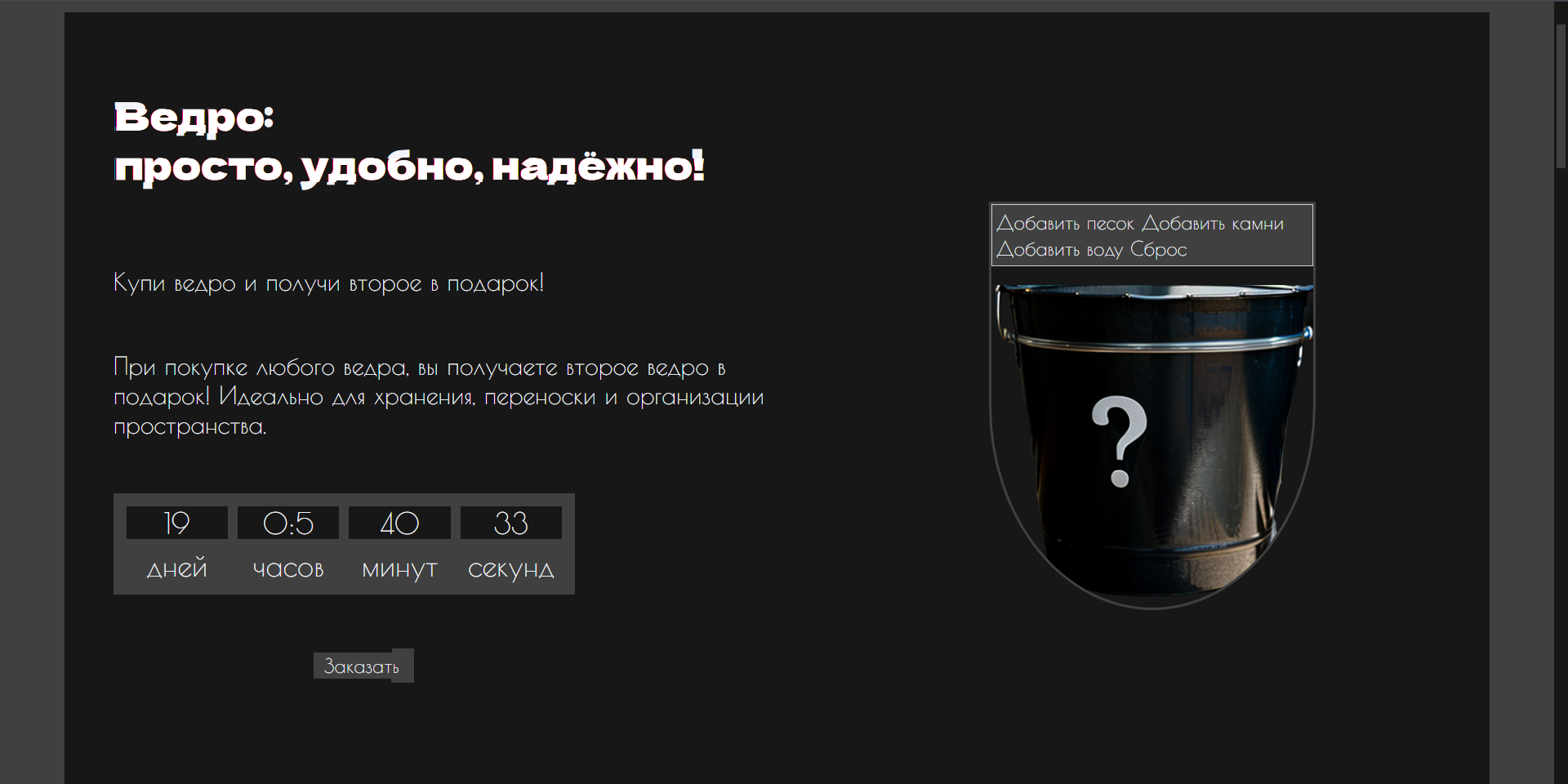
**Обоснование:** Эффект «глюка» привлекает внимание к заголовку и создает впечатление современности и технологичности. Таймер до окончания акции создает ощущение срочности и мотивирует пользователей к совершению целевого действия (регистрации или покупке курса). Интерактивное ведро — оригинальное решение, привлекающее внимание и позволяющее пользователю взаимодействовать с контентом, что повышает его вовлеченность и запоминаемость.****

Рисунок 3. Баннер

function addParticles(type, cost) {

  if(parseInt(getComputedStyle(bucket).height)>0){

    const bucketRect = bucket.getBoundingClientRect();

    const size = getRandomFloat(particleSizes[type].min, particleSizes[type].max);

    for (let i = 0; i < cost; i++) {

      const particle = document.createElement('div');

      particle.classList.add('particleFirst');

      particle.style.backgroundColor = particleColors[type][Math.floor(Math.random() \* particleColors[type].length)];

      particle.style.width = size + 'px';

      particle.style.height = size + 'px';

      let x = getRandomFloat(bucketRect.left + size, bucketRect.right - size);

      let startY = bucketRect.top;

      let endY = bucketRect.bottom - size;

      particle.style.left = x - containerBucket.getBoundingClientRect().left + 'px';

      particle.style.top = startY - containerBucket.getBoundingClientRect().top + 'px';

      particle.style.zIndex = 1;

      particle.style.opacity = 0;

      containerBucket.appendChild(particle);

      const newParticle = {

        element: particle,

        x: x - containerBucket.getBoundingClientRect().left,

        y: startY - containerBucket.getBoundingClientRect().top,

        size: size,

        type: type

      };

      if (newParticle.element) {

        particles.push(newParticle);

        const TimeFALL = 1.5 \* particleDensities[type];

        newParticle.element.style.transition = `top ${TimeFALL}s ease-in-out, opacity ${TimeFALL / 2}s ease-in`;

        setTimeout(() => {

          if (newParticle.element) {

            newParticle.element.style.top = endY - containerBucket.getBoundingClientRect().top + 'px';

            newParticle.element.style.opacity = 0.8;

          }

        }, 10);

        setTimeout(() => {

          if (newParticle.element) {

            newParticle.element.style.transition = 'none';

          }

        }, TimeFALL \* 1000);

      }

    }

    bucket.style.height=bucket.offsetHeight-size>0?bucket.offsetHeight-size + "px":"0px";

    bucket.style.bottom=parseInt(getComputedStyle(bucket).bottom)+size-1 + "px";

  }

}

**1.3. Блок “О продукте”**

Блок «О продукте» предназначен для подробного ознакомления пользователя с содержанием и преимуществами онлайн-курса по веб-дизайну. В данном проекте блок реализован **в виде интерактивных карточек**. При нажатии на карточку ее лицевая сторона уходит вниз и плано исчезает, открывая подробную информацию о соответствующем аспекте ведра.

**Технологии:** HTML, CSS (включая CSS-переходы или анимацию для эффекта ухода карточки), JavaScript (для обработки кликов по карточкам и управления анимацией).

**Интерактивные элементы и анимационные эффекты:**

* **Интерактивные карточки (HTML, CSS, JavaScript):** на странице отображается несколько карточек, каждая из которых представляет собой отдельный модуль или тему курса. При нажатии на карточку происходит анимация ухода, раскрывающая подробное описание ведра, его характеристики и преимушества, Карточки содержат изображения иллюстрирующие содержание текста.
* **Плавная анимация (CSS-переходы/анимации):** анимация ухода карточки реализована с помощью **JavaScript**, что обеспечивает плавный и визуально привлекательный эффект.

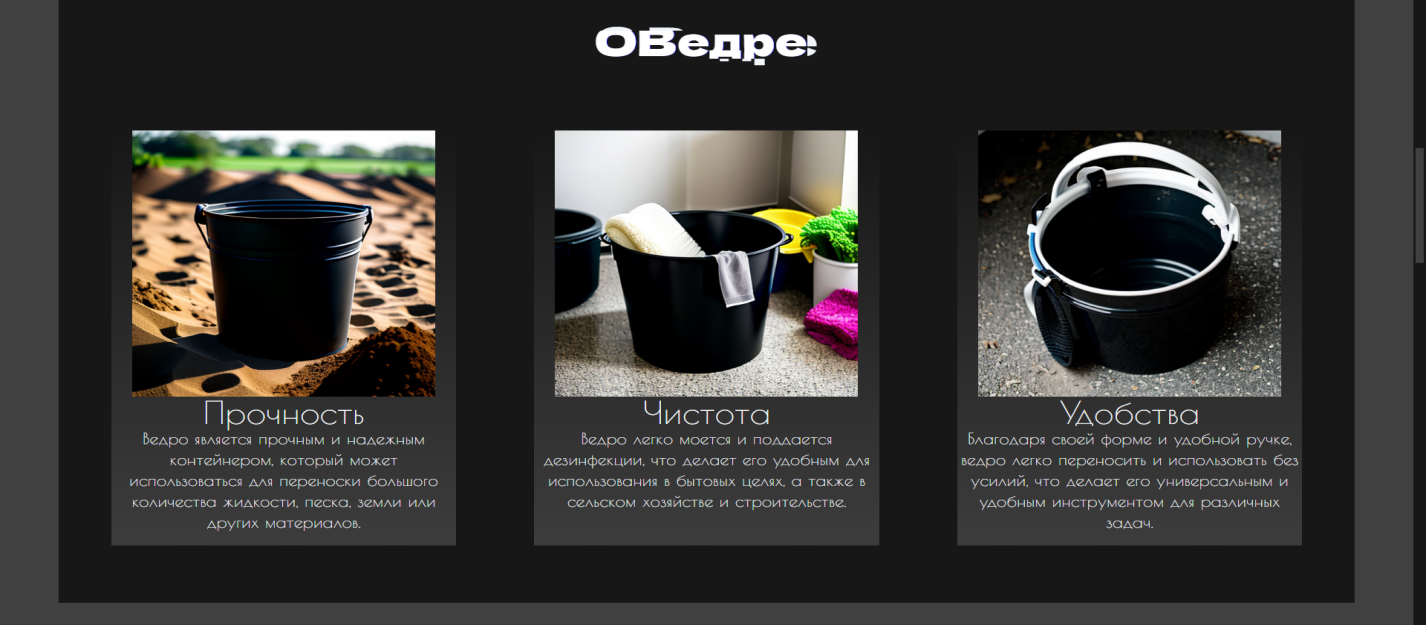
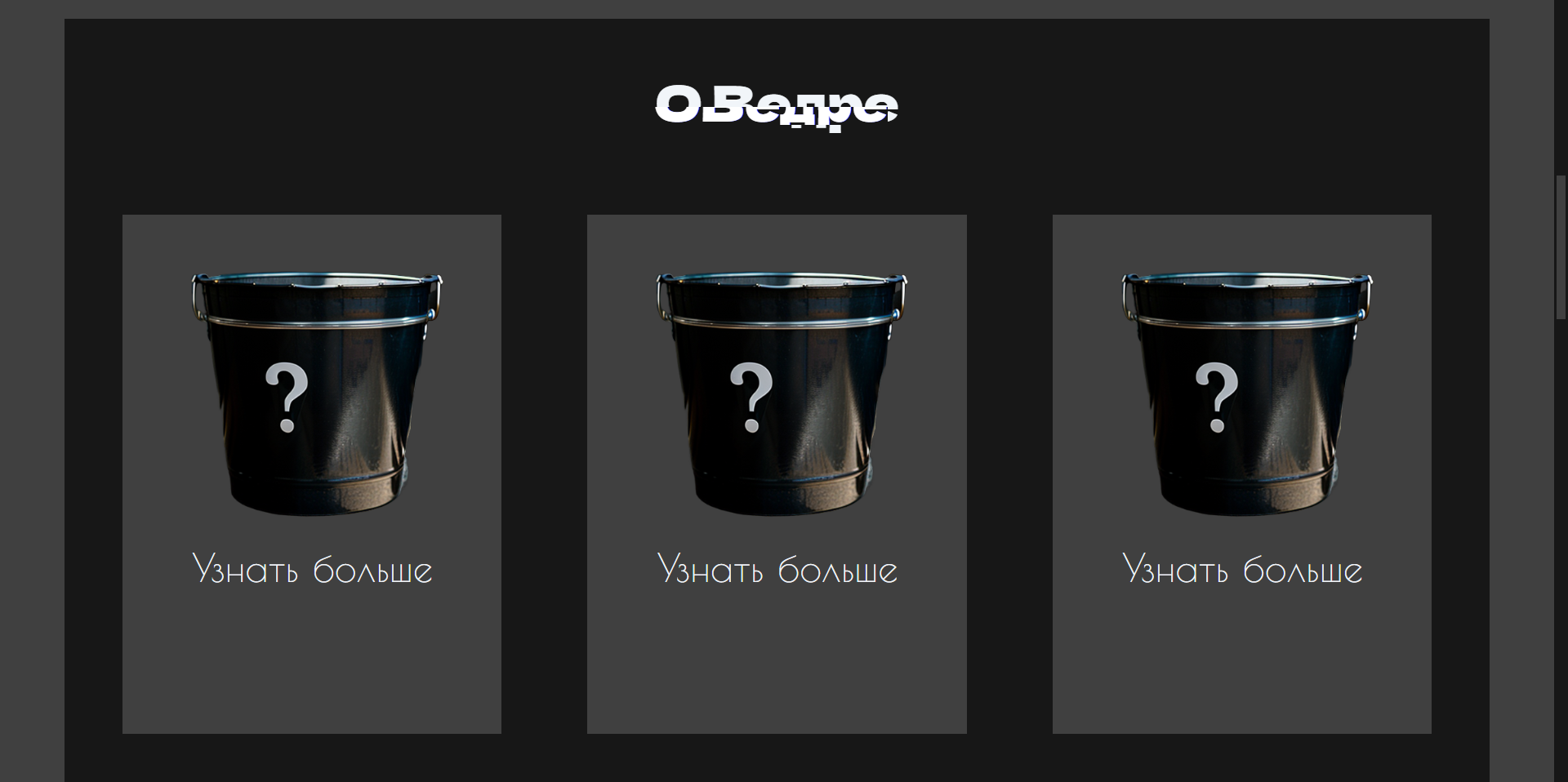
**Обоснование:** Использование интерактивных карточек позволяет структурировать информацию о ведре и представить ее в удобной и увлекательной форме. Анимация открытия карточки привлекает внимание и делает взаимодействие с блоком более интересным. Такая организация информации помогает пользователю быстро получить представление о ведре и его преимуществах.****

Рисунок 4. Раскрытые карточки

****Рисунок 5. Раскрытые карточки

**Код Исчезновения обертки:**

function ChoiseDisappear(afterDisappear, index, array) {

    this.classList.add('activeInfo');

    setTimeout(() => {

        this.classList.add('hidden');

        afterDisappear.classList.remove('hidden');

        afterDisappear.classList.remove('activeInfo');

        if (window.matchMedia("(max-width: 1024px)").matches) {

            index >= array.length - 1 ? index = 1 : index += 2;

            afterDisappear.addEventListener('click', ChoiseDisappear.bind(afterDisappear, array[index], index, array))

        };

        Linear\_Bg(afterDisappear, false);

    }, 1000);

}

**1.4. Блок “каталог”**

Раздел «Каталог» представляет собой визуальную демонстрацию предлагаемых товаров. В данном проекте «Каталог» реализован **в виде скроллера**, в котором товары (вёдра) плавно сменяют друг друга. Каждый товар появляется с эффектом постепенного проявления, а фон карточки товара постепенно меняется с черного на линейный градиент от черного к серому.

**Технологии:** HTML, CSS (включая CSS-переходы и анимацию), JavaScript (для управления скроллером, анимацией появления и смены фона).

**Интерактивные элементы и анимационные эффекты:**

* **скроллер (HTML, CSS, JavaScript):** товары отображаются в виде горизонтальной полосы.
* **Эффект плавного исчезновения и появления (CSS-переходы/анимации и JavaScript):** при смене товара текущий товар плавно исчезает (прозрачность уменьшается до 0), а следующий товар плавно появляется (прозрачность увеличивается от 0 до 1).
* **Анимация изменения фона (CSS):** фон карточки товара плавно переходит от черного к линейному градиенту от черного к серому, когда товар появляется на экране.

**Обоснование:** Скроллер позволяет компактно представить большое количество товаров. Эффекты плавного исчезновения и появления делают просмотр каталога более динамичным и интересным. Анимация изменения фона привлекает внимание к каждому товару и создает визуальную связь между товарами в каталоге.

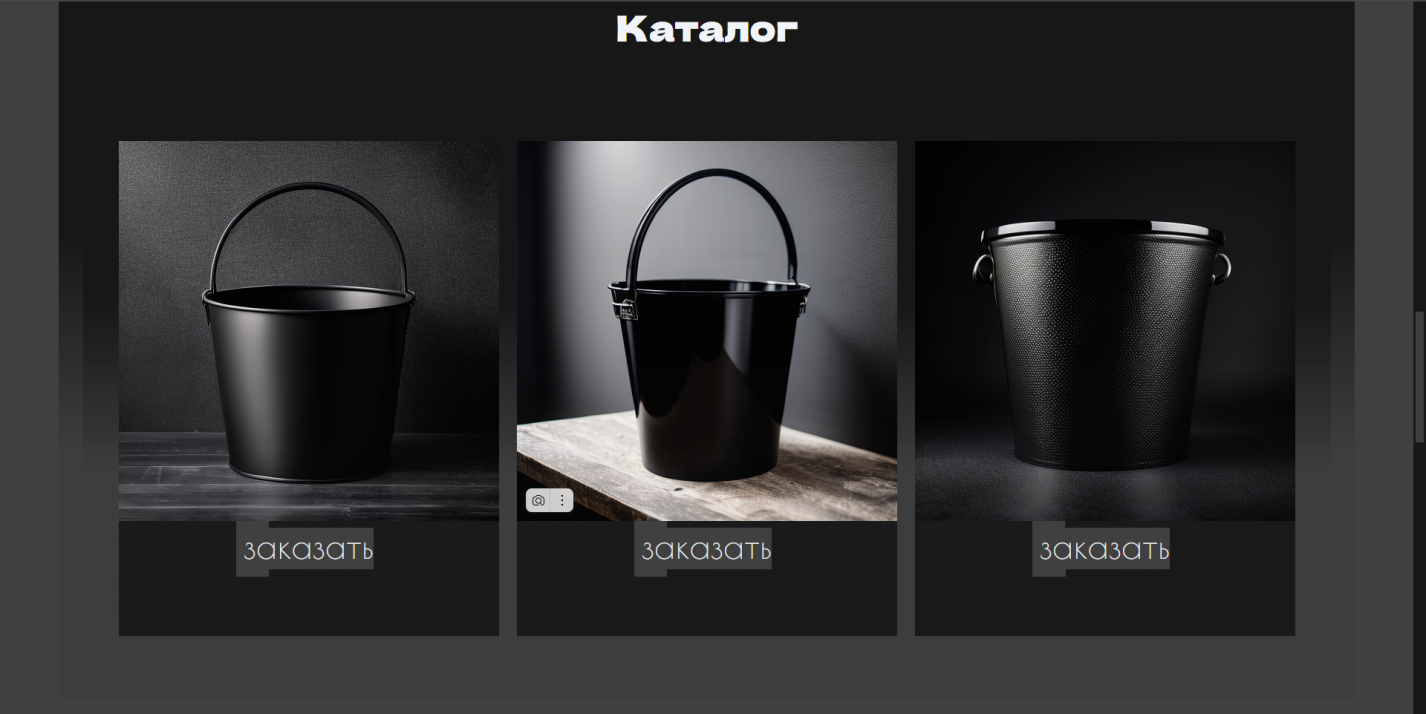
****

Рисунок 6. Скроллер каталога

**1.5. Блок “Комментарии”**

Блок «Комментарии» предназначен для сбора отзывов и мнений пользователей об ведре. В данном проекте блок включает в себя следующие элементы: возможность добавления комментария через всплывающее меню, отображение существующих комментариев в виде слайдера, а также анимацию исчезновения и появления текста при пролистывании комментариев.

**Технологии:** HTML, CSS (включая CSS-анимацию), JavaScript (для управления всплывающим меню, слайдером, анимацией текста и отправкой комментариев).

**Интерактивные элементы и анимационные эффекты:**

* **Всплывающее меню для добавления комментария (HTML, CSS, JavaScript):** кнопка «Добавить комментарий» вызывает всплывающее модальное окно, в котором пользователь может ввести текст и указать свое имя и аватарку(не обязательно). После отправки комментарий отправляется на сервер и отображается в слайдере.
* **Слайдер комментариев (HTML, CSS, JavaScript):** комментарии отображаются в виде слайдера, позволяющего пользователю просматривать отзывы других пользователей.
* **Анимация исчезновения и появления текста (CSS-анимации и JavaScript):** при пролистывании комментария текст текущего комментария постепенно «разлетается» на отдельные буквы и исчезает. Затем текст следующего комментария появляется в случайных местах на экране, а после короткой задержки собирается в единый текст с помощью плавной анимации.

**Обоснование:** всплывающее меню упрощает процесс добавления комментария. Слайдер позволяет компактно отображать большое количество отзывов. Анимация исчезновения и появления текста делает просмотр комментариев более интересным и запоминающимся, а также привлекает внимание к содержанию отзывов.

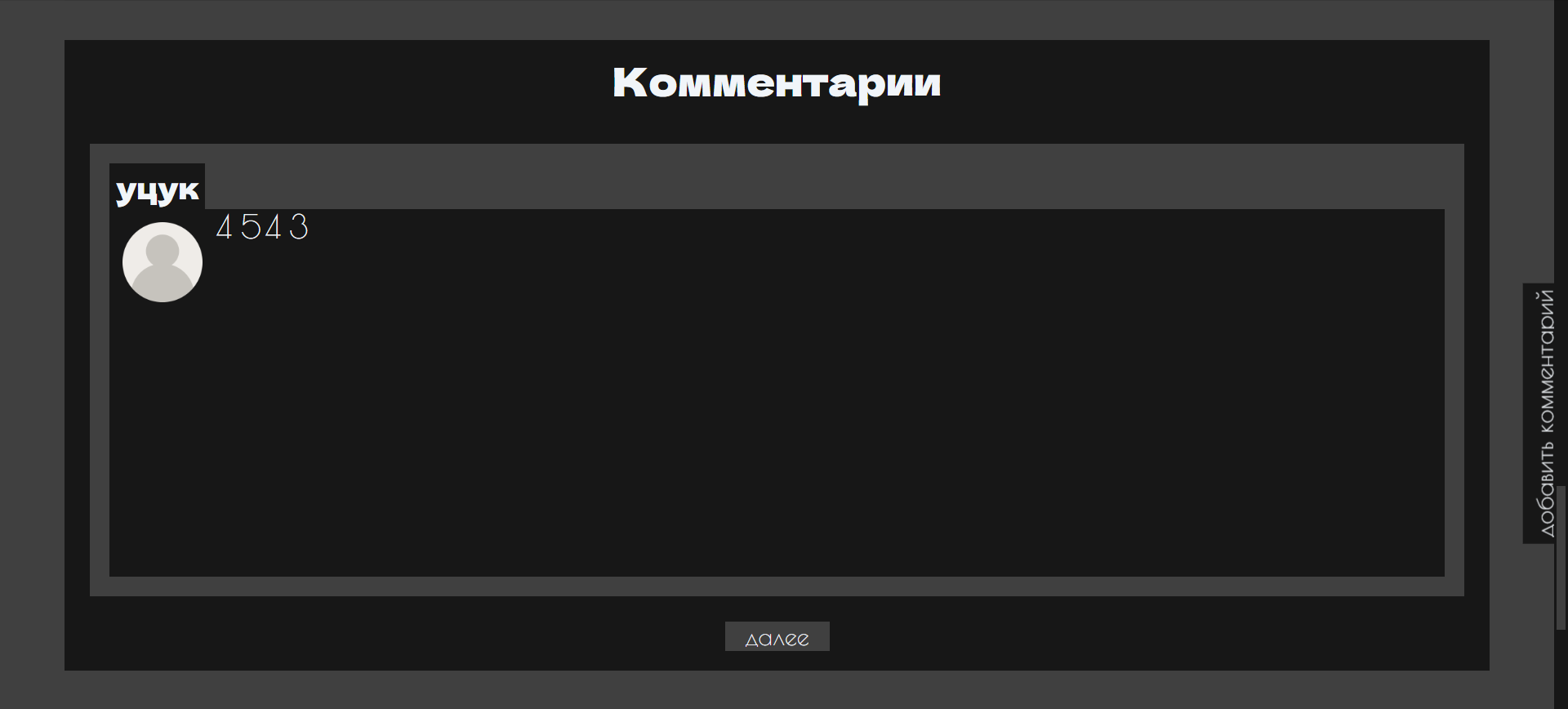


Рисунок 7. Скроллер каталога

**1.6. Блок “Обратная связь”**

Блок «Обратная связь» предназначен для сбора контактных данных пользователей, заинтересованных в ведре, и предоставления им возможности получить скидку. В данном проекте блок включает в себя следующие элементы: форму для заказа «ведра» с указанием имени и электронной почты, отправку подтверждения заказа на электронную почту, а также интерактивную «стираемую» карточку, под которой скрывается трехмерный параллелепипед, показывающий случайную скидку.

**Технологии:** HTML, CSS, JavaScript (для обработки формы, анимации параллелепипеда и отправки данных), Canvas API (для реализации стираемой области), Node.js(express,nodemailer) (для обработки запросов на сервере и отправки электронных писем).

**Интерактивные элементы и анимационные эффекты:**

* **Форма заказа «ведра» (HTML):** пользователь вводит свое имя и адрес электронной почты в соответствующие поля и нажимает кнопку «Заказать ведро».
* **Отправка подтверждения заказа по электронной почте (JavaScript, Node.js(**express,nodemailer**)):** после отправки формы данные пользователя отправляются на сервер. Сервер генерирует и отправляет письмо с подтверждением заказа на указанный адрес электронной почты.
* **Стираемая карточка (HTML, CSS, JavaScript, Canvas API):** часть экрана закрыта «стираемой» областью, имитирующей лотерейный билет. Пользователь должен «стереть» эту область, чтобы увидеть скрытый под ней параллелепипед.
* **Трехмерный анимированный параллелепипед со скидкой (HTML, CSS, JavaScript):** под стираемой карточкой находится трехмерный параллелепипед. При нажатии на него он начинает вращаться, а затем случайным образом «падает» одной из шести граней вверх. На каждой грани указан размер скидки (от 0% до 30%). Выпавшая грань отображает скидку, которую получает пользователь при заказе курса.

**Обоснование:** Форма заказа «ведра» позволяет собрать контактные данные заинтересованных пользователей. Отправка подтверждения заказа создает положительное впечатление и обеспечивает обратную связь. Интерактивная «стираемая» карточка и анимированный параллелепипед добавляют элемент игры и азарта, делая процесс получения скидки более увлекательным.

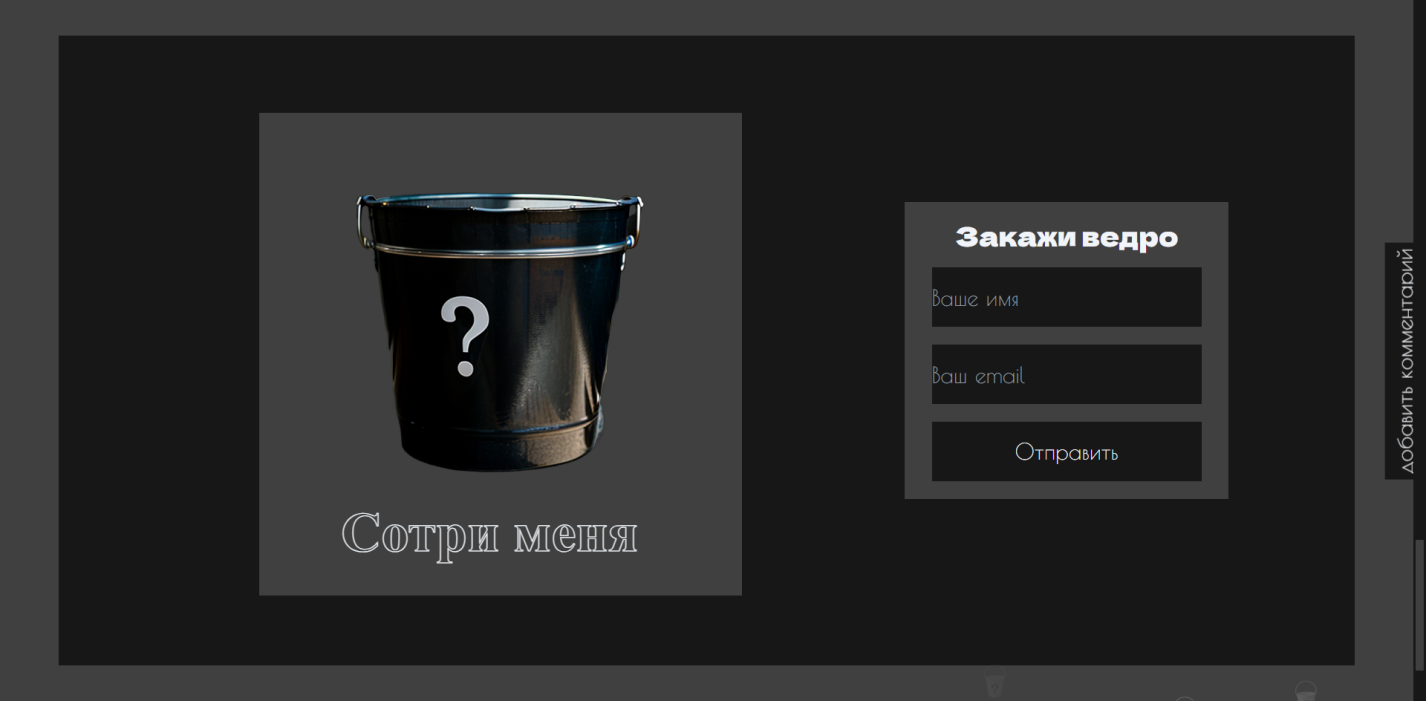


Рисунок 8. Обратная связь

Код стирания карточки:

let canvas = document.getElementById("canvas");

if (canvas.getContext) {

let ctx = canvas.getContext("2d");

ctx.clearRect(0, 0, 800, 800);

ctx.fillStyle = window.getComputedStyle(document.querySelector('body')).getPropertyValue('background-color');

ctx.strokeStyle = window.getComputedStyle(document.querySelector('body')).getPropertyValue('color');

ctx.font = 'bold 75px Dela Gothic One serif';

if ((window.matchMedia("(max-width: 768px)").matches)) {

ctx.fillRect(80, 80, 650, 650);

ctx.strokeText("Сотри меня", 190, 670);

}

else {

ctx.fillRect(100, 80, 650, 650);

ctx.strokeText("Сотри меня", 210, 670);

}

const img = new Image();

img.src = "http://localhost:3000/images/image1.png";

img.onload = function () {

const pattern = ctx.createPattern(img, "no-repeat");

ctx.fillStyle = pattern;

if ((window.matchMedia("(max-width: 1024px)").matches)) {

ctx.translate(140, 100);

}

else {

ctx.translate(160, 100);

}

ctx.fillRect(0, 0, 650, 650);

};

if (!window.innerWidth < 820) {

canvas.addEventListener('mousemove', (event) => ctx.clearRect(event.offsetX - 160, event.offsetY - 160, 80, 80));

}

canvas.onclick = function () {

let y = -30;

let DisappearCanvas = setInterval(() => {

ctx.clearRect(-100, y += 2, 800, 20);

y >= 800 ? (clearInterval(DisappearCanvas), canvas.style.zIndex = -3) : null;

}, 1)

}

}

**1.7. Блок “Подвал сайта”**

Блок «Футер» содержит информацию об авторских правах, контактные данные (телефон и электронную почту) и служит визуальным завершением целевой страницы. В данном проекте футер включает в себя анимацию: сверху падают вёдра, которые при соприкосновении с нижней границей футера взрываются, рассыпаясь на пиксели.

**Технологии:** HTML, CSS (включая CSS-анимацию), JavaScript (для управления анимацией падения и взрыва вёдер, а также динамическим созданием элементов).

**Интерактивные элементы и анимационные эффекты:**

* **Падающие вёдра (HTML, CSS-анимация, JavaScript):** Вёдра создаются динамически в верхней части футера (за его пределами). Им задаются случайные значения горизонтальной позиции. CSS-анимация используется для плавного падения вёдер вниз.
* **Взрыв ведер (CSS-анимация, JavaScript):** при соприкосновении с нижней границей футера (определяется с помощью JavaScript) ведро удаляется, и на его месте создается множество маленьких элементов (пикселей). Этим пикселям задаются случайные направления и скорости разлета. CSS-анимация используется для анимации разлета и постепенного исчезновения пикселей.

**Обоснование:** Анимация падающих и взрывающихся вёдер привлекает внимание к футеру и создаёт запоминающийся визуальный эффект. Она может быть интерпретирована как символ «разрушения старых знаний» или «взрывного обучения».

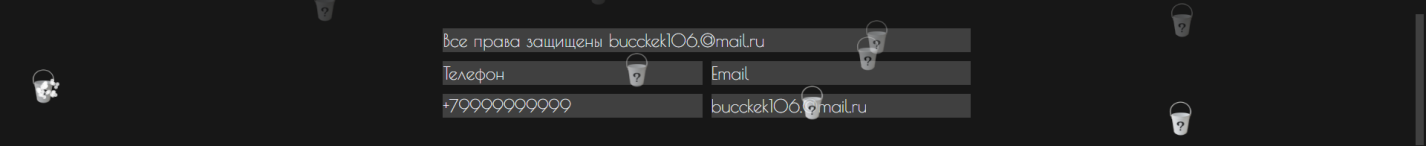


Рисунок 9. Футер

**ЗАКЛЮЧЕНИЕ**

В рамках данной курсовой работы была разработана целевая страница, посвященная теме «Ведро». Целью работы было создание целевой страницы, которая не только эффективно представляет концепцию «ведра», но и побуждает пользователей заказать ведро.

В процессе исследования были решены следующие задачи:

* Определена целевая аудитория и её потребности, связанные с темой «Ведра».
* Разработана уникальная концепция целевой страницы, в которой «ведро» является ключевым визуальным и смысловым элементом.
* Созданы отдельные блоки Landing Page, объединённые общей темой «Ведра»: «Шапка сайта», «Баннер», «О продукте», «Комментарии», «Обратная связь» и «Футер», каждый из которых усиливает концепцию «Ведра».
* Для каждого блока были реализованы оригинальные дизайнерские и программные решения, ориентированные на привлечение внимания и создание запоминающегося опыта взаимодействия с целевой страницей.
* Продемонстрировано применение современных веб-технологий для реализации интерактивных и вовлекающих элементов, связанных с темой «Ведра».

Наиболее удачными и концептуально важными решениями, реализованными в данной работе, являются:

* **Интерактивное ведро в баннере:** позволяет пользователю почувствовать себя активным участником процесса, «наполняя» ведро песком, камнями или водой.
* **Стираемая карточка и анимированный параллелепипед в блоке «Обратная связь»:** предлагают пользователю «выиграть» скидку на «Ведро» или набор, связанный с ним, добавляя элемент игры и азарта.
* **Анимация падающих и взрывающихся вёдер в футере:** создаёт визуально яркий и запоминающийся эффект, символизирующий «разрушение стереотипов» или «взрыв идей» при использовании «Вёдра».

Разработанная целевая страница демонстрирует возможность создания оригинального и эффективного веб-интерфейса, основанного на единой концепции «Ведра». Визуальные и интерактивные элементы целевой страницы направлены на привлечение внимания к теме «Ведра» и стимулирование пользователей к целевому действию.

В заключение следует подчеркнуть, что успех целевой страницы во многом зависит от чёткой концепции и последовательной реализации всех элементов, объединённых общей темой. Данная курсовая работа представляет собой пример такого подхода, где «ведро» выступает не просто в качестве графического элемента, а является ключевым смысловым ядром всей целевой страницы.

СПИСОК [ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ](#_Toc515549778)

1. Руководство по express доступная по адресу [**https://metanit.com/web/nodejs/4.1.php**](https://metanit.com/web/nodejs/4.1.php)
2. Документация по nodemailer доступная по адресу [**https://nodejsdev.ru/guides/webdraftt/email/**](https://nodejsdev.ru/guides/webdraftt/email/)
3. Документация по tailwindcss доступная по адресу [**https://tailwindcss.com/**](https://tailwindcss.com/%20)
4. Документация по sqlite доступная по адресу [**https://nodejs.org/api/sqlite.html**](https://nodejs.org/api/sqlite.html)
5. Документация по path доступная по адресу [**https://nodejs.org/api/path.html**](https://nodejs.org/api/path.html%20)
6. Документация по Node.js доступная по адресу [**https://nodejs.org/docs/latest/api**](https://nodejs.org/docs/latest/api)**/**
7. Документация по HTML доступная по адресу [**https://developer.mozilla.org/ru/docs/Web/HTML**](https://developer.mozilla.org/ru/docs/Web/HTML)
8. Документация по CSS доступная по адресу [**http://developer.mozilla.org/en-US/docs/Web/CSS**](http://developer.mozilla.org/en-US/docs/Web/CSS)
9. Документация по JavaScript доступная по адресу [**https://developer.mozilla.org/ru/docs/Web/JavaScript**](https://developer.mozilla.org/ru/docs/Web/JavaScript)
10. Документация по Canvas\_API доступная по адресу [**https://developer.mozilla.org/en-US/docs/Web/API/Canvas\_API**](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API)