

Санкт-Петербургский Политехнический Университет Петра Великого  
Физико-Механический институт

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5**

**«Генерация лендинга с помощью ИИ»**

**По дисциплине «Автоматизация научных исследований»**

Выполнил

студент гр. № 5040102/50201

Усманов И. Р.

Преподаватель:

Новиков Ф. А.

Санкт-Петербург

2025 г.

## СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>3</b>
<b>Цель .....</b>	<b>3</b>
<b>Исходные данные .....</b>	<b>3</b>
<b>Порядок выполнения .....</b>	<b>3</b>
<b>Форма отчёта.....</b>	<b>4</b>
<b>ПРОМПТ ДЛЯ ГЕНЕРАЦИИ .....</b>	<b>5</b>
<b>МОДЕЛЬ .....</b>	<b>7</b>
<b>АНАЛИЗ.....</b>	<b>8</b>
<b>ВЫВОДЫ.....</b>	<b>28</b>

## **ВВЕДЕНИЕ**

### **Цель**

Провести эксперимент по автоматической генерации одностораничного веб-сайта (лендинга) с использованием ИИ-инструмента, оценить качество результата по функциональным и техническим критериям, а также измерить временные затраты на выполнение задачи.

### **Исходные данные**

1. Промпт для генерации — полное текстовое описание лендинга, включающее:

- название продукта/проекта;
- краткое описание (цель, преимущества, целевая аудитория);
- требуемые блоки (например: заголовок, описание, функции, отзывы, форма обратной связи, футер);
- предпочтения по стилю (например: минимализм, корпоративный стиль, акцент на call-to-action).

2. Технические требования — правила для ИИ-системы:

- использовать только HTML, CSS и JavaScript (без внешних зависимостей) или разрешить конкретные библиотеки (например, Bootstrap);
- обеспечить адаптивность (корректное отображение на мобильных устройствах);
- соблюдать принципы модульности и читаемости кода;
- обеспечить кроссплатформенную совместимость (работоспособность в любой ОС и современном браузере).

3. Описание ИТ-проекта — реальный или условный проект, для которого создаётся лендинг

### **Порядок выполнения**

1. Подготовьте промпт, технические требования и описание проекта.
2. Передайте эти данные выбранному ИИ-инструменту (например, Cursor, GitHub Copilot, Claude и др.).

3. Запросите генерацию полноценного лендинга в виде одного или нескольких файлов (index.html, style.css, script.js).

4. Зафиксируйте время начала и окончания генерации (в минутах).

5. Проверьте результат по следующим критериям (ответ — да или нет):

- Лендинг содержит все запрошенные блоки — ;
- Код валиден и не содержит синтаксических ошибок — ;
- Страница корректно отображается в браузере — ;
- Реализована адаптивная вёрстка — ;
- Код легко читаем и подходит для повторного использования — .

6. По шкале от 1 до 5 оцените:

- соответствие дизайна и содержания исходному описанию проекта;
- техническое качество клиентского кода;
- общую пригодность лендинга для практического применения.

### **Форма отчёта**

Отчёт оформляется в виде PDF-документа и должен включать:

- краткое описание ИТ-проекта;
- текст промпта и технических требований;
- зафиксированное время выполнения (в минутах);
- заполненную таблицу с ответами «да/нет» и оценками по шкале;
- вывод о целесообразности использования ИИ для генерации лендингов

в научно-технических и прикладных задачах.

Сгенерированные файлы лендинга сохраняются в архив или публикуются в репозитории на GitHub.

## **ПРОМПТ ДЛЯ ГЕНЕРАЦИИ**

### **Исходные данные**

Создай полноценный, адаптивный одностраничный сайт (лендинг) для проекта.

Название проекта: Система заказа еды в ресторане

Краткое описание: цель проекта - обеспечить посетителям ресторана удобный и быстрый способ самостоятельного оформления заказа без участия официанта. Задачи проекта: реализовать цифровое меню с актуальной информацией о блюдах. обеспечить возможность оформления и изменения заказа через пользовательский интерфейс. Реализовать отображение текущего статуса заказа. Организовать передачу заказа на кухню и инструменты его обработки сотрудниками. Обеспечить управление меню и параметрами блюд со стороны персонала ресторана.

Требуемые блоки: Шапка (логотип “MUCHACHO”, навигация по сайту), блок с авторизацией пользователя, блок с меню, где сверху будут указаны категории блюд (стартеры, салаты, супы, десерты, напитки). Несколько демо-позиций для каждой категории. Возможность добавлять и убавлять количество позиций в главном меню. Блок с корзиной, в которой будут отображаться выбранные позиции.

Стиль: яркий, мексиканская тематика, зазывающий. Главный акцент на привлечении внимания. Цветовая палитра: коричневый, белый, желтый, зеленый.

### **Технические требования**

использовать только HTML, CSS и JavaScript (без внешних зависимостей) или разрешить конкретные библиотеки (например, Bootstrap);

обеспечить адаптивность (корректное отображение на мобильных устройствах);

соблюдать принципы модульности и читаемости кода;

обеспечить кроссплатформенную совместимость (работоспособность в любой ОС и современном браузере)

Комментарии в коде

Плавная прокрутка

## **МОДЕЛЬ**

Gemini 3

## АНАЛИЗ

Время выполнения – 5 минут с учетом уточняющего промпта

### Результат

Конечно. Ниже представлен полный код для создания адаптивного одностраничного сайта (лендинга) для системы заказа еды "MUCHACHO".

Я разделил код на три части (HTML, CSS, JS) для удобства чтения и соблюдения принципов модульности. Вы можете сохранить их в отдельные файлы (`index.html`, `style.css`, `script.js`) или объединить в один файл.

Особенности реализации:

- **Стиль:** Яркий, с использованием теплых "мексиканских" цветов (желтый фон, зеленые кнопки, коричневые акценты).
- **Технологии:** Чистый HTML5, CSS3 (Flexbox/Grid), Vanilla JavaScript (ES6+).
- **Функционал:** Фильтрация меню по категориям, добавление/удаление товаров, динамическая корзина, подсчет итоговой суммы.
- **Адаптивность:** Полная поддержка мобильных устройств.

---

#### 1. HTML (`index.html`)

Структура страницы с семантическими тегами.

HTML

```
<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>MUCHACHO | Мексиканская кухня</title>
    <link rel="stylesheet" href="style.css">
    <link
        href="https://fonts.googleapis.com/css2?family=Lobster&family=Roboto:wght@400
        ;700&display=swap" rel="stylesheet">
</head>
<body>

<header class="header">
    <div class="container header-content">
        <div class="logo">MUCHACHO <img alt="Logo icon" /></div>
        <nav class="nav">
            <a href="#auth">Войти</a>
            <a href="#menu">Меню</a>
            <a href="#cart">Корзина <span id="cart-count">(0)</span></a>
        </nav>
    </div>
</header>

<section id="auth" class="hero-section">
    <div class="container">
        <div class="hero-card">
```

```

        <h1>Hola, Amigo!</h1>
        <p>Добро пожаловать в Muchacho. Заказывай быстро, ешь
вкусно!</p>
        <form class="auth-form" onsubmit="event.preventDefault();
alert('Вы авторизованы!');">
            <input type="text" placeholder="Ваше имя" required>
            <input type="number" placeholder="Номер столика" min="1"
max="20" required>
            <button type="submit" class="btn btn-primary">Начать
заказ</button>
        </form>
    </div>
</div>
</section>

<section id="menu" class="menu-section">
    <div class="container">
        <h2 class="section-title">Наше Меню</h2>

        <div class="menu-tabs">
            <button class="tab-btn active" data-
category="all">Все</button>
            <button class="tab-btn" data-
category="starters">Стarterы</button>
            <button class="tab-btn" data-
category="salads">Салаты</button>
            <button class="tab-btn" data-category="soups">Супы</button>
            <button class="tab-btn" data-
category="desserts">Десерты</button>
            <button class="tab-btn" data-
category="drinks">Напитки</button>
        </div>

        <div id="menu-grid" class="menu-grid">
        </div>
    </div>
</section>

<section id="cart" class="cart-section">
    <div class="container">
        <h2 class="section-title">Ваш Заказ</h2>
        <div class="cart-wrapper">
            <div id="cart-items" class="cart-items">
                <p class="empty-cart-msg">Корзина пуста, амиго. Добавь
немного огня!</p>
            </div>
            <div class="cart-summary">
                <div class="total-price">Итого: <span id="total-
amount">0</span> ₽</div>
                <button class="btn btn-success" onclick="alert('Заказ
отправлен на кухню! Ожидайте.')">Оформить заказ</button>
            </div>
        </div>
    </div>
</section>

<footer class="footer">
    <div class="container">
        <p>&copy; 2023 Restaurant Muchacho. Buen provecho!</p>
    </div>
</footer>

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

```

```
</body>
</html>
2. CSS (style.css)
```

Оформление в мексиканском стиле: яркость, контраст, скругления.

## CSS

```
/* --- Сброс и Базовые настройки --- */
:root {
    --color-brown: #5D4037;
    --color-white: #FFFFFF;
    --color-yellow: #FFC107; /* Яркий желтый */
    --color-yellow-light: #FFF8E1;
    --color-green: #2E7D32; /* Насыщенный зеленый */
    --color-red: #D32F2F;
    --font-heading: 'Lobster', cursive;
    --font-body: 'Roboto', sans-serif;
}

* {
    box-sizing: border-box;
    margin: 0;
    padding: 0;
}

html {
    scroll-behavior: smooth; /* Плавная прокрутка */
}

body {
    font-family: var(--font-body);
    background-color: var(--color-yellow-light);
    color: var(--color-brown);
    line-height: 1.6;
}

.container {
    max-width: 1200px;
    margin: 0 auto;
    padding: 0 20px;
}

/* --- Кнопки --- */
.btn {
    padding: 10px 20px;
    border: none;
    border-radius: 25px;
    cursor: pointer;
    font-weight: bold;
    transition: transform 0.2s, box-shadow 0.2s;
    font-family: var(--font-body);
}

.btn:hover {
    transform: translateY(-2px);
    box-shadow: 0 4px 8px rgba(0,0,0,0.2);
}

.btn-primary {
    background-color: var(--color-brown);
    color: var(--color-white);
```

```

}

.btn-success {
    background-color: var(--color-green);
    color: var(--color-white);
    font-size: 1.2rem;
    width: 100%;
}

/* --- Шапка --- */
.header {
    background-color: var(--color-brown);
    color: var(--color-yellow);
    padding: 15px 0;
    position: sticky;
    top: 0;
    z-index: 1000;
    box-shadow: 0 2px 10px rgba(0,0,0,0.3);
}

.header-content {
    display: flex;
    justify-content: space-between;
    align-items: center;
}

.logo {
    font-family: var(--font-heading);
    font-size: 2rem;
}

.nav a {
    color: var(--color-white);
    text-decoration: none;
    margin-left: 20px;
    font-weight: bold;
    transition: color 0.3s;
}

.nav a:hover {
    color: var(--color-yellow);
}

/* --- Секция Авторизации (Hero) --- */
.hero-section {
    background: linear-gradient(135deg, var(--color-yellow) 0%, #FF9800 100%);
    padding: 80px 0;
    text-align: center;
}

.hero-card {
    background: var(--color-white);
    padding: 40px;
    border-radius: 15px;
    display: inline-block;
    box-shadow: 0 10px 20px rgba(0,0,0,0.1);
    border: 2px solid var(--color-brown);
    max-width: 500px;
    width: 100%;
}

.hero-card h1 {

```

```

        font-family: var(--font-heading);
        color: var(--color-green);
        font-size: 3rem;
        margin-bottom: 10px;
    }

.auth-form {
    display: flex;
    flex-direction: column;
    gap: 15px;
    margin-top: 20px;
}

.auth-form input {
    padding: 12px;
    border: 2px solid #ddd;
    border-radius: 8px;
    font-size: 1rem;
}

/* --- Меню --- */
.menu-section {
    padding: 60px 0;
}

.section-title {
    text-align: center;
    font-family: var(--font-heading);
    font-size: 2.5rem;
    color: var(--color-brown);
    margin-bottom: 30px;
}

.menu-tabs {
    display: flex;
    justify-content: center;
    flex-wrap: wrap;
    gap: 10px;
    margin-bottom: 40px;
}

.tab-btn {
    background: transparent;
    border: 2px solid var(--color-brown);
    padding: 8px 16px;
    border-radius: 20px;
    cursor: pointer;
    font-weight: bold;
    color: var(--color-brown);
}

.tab-btn.active, .tab-btn:hover {
    background-color: var(--color-green);
    color: var(--color-white);
    border-color: var(--color-green);
}

.menu-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));
    gap: 30px;
}

```

```
.menu-item {
    background: var(--color-white);
    border-radius: 15px;
    overflow: hidden;
    box-shadow: 0 5px 15px rgba(0,0,0,0.05);
    transition: transform 0.3s;
    border-bottom: 5px solid var(--color-green);
}

.menu-item:hover {
    transform: translateY(-5px);
}

/* Используем плейсхолдеры вместо реальных картинок */
.item-img {
    width: 100%;
    height: 200px;
    background-color: #eee;
    object-fit: cover;
}

.item-content {
    padding: 20px;
}

.item-title {
    font-size: 1.25rem;
    font-weight: bold;
    margin-bottom: 5px;
}

.item-desc {
    font-size: 0.9rem;
    color: #666;
    margin-bottom: 15px;
    min-height: 40px;
}

.item-footer {
    display: flex;
    justify-content: space-between;
    align-items: center;
}

.item-price {
    font-size: 1.2rem;
    font-weight: bold;
    color: var(--color-brown);
}

.item-controls {
    display: flex;
    align-items: center;
    gap: 10px;
}

.qty-btn {
    background: var(--color-brown);
    color: white;
    width: 30px;
    height: 30px;
    border-radius: 50%;
    border: none;
}
```

```

        cursor: pointer;
        font-size: 1.2rem;
        display: flex;
        justify-content: center;
        align-items: center;
    }

/* --- Корзина --- */
.cart-section {
    padding: 60px 0;
    background-color: var(--color-white);
    border-top: 2px solid var(--color-brown);
}

.cart-wrapper {
    max-width: 800px;
    margin: 0 auto;
}

.cart-item {
    display: flex;
    justify-content: space-between;
    align-items: center;
    padding: 15px;
    border-bottom: 1px solid #eee;
}

.cart-item-title {
    font-weight: bold;
    flex: 1;
}

.cart-summary {
    margin-top: 30px;
    text-align: right;
    padding: 20px;
    background: var(--color-yellow-light);
    border-radius: 10px;
}

.total-price {
    font-size: 1.5rem;
    font-weight: bold;
    margin-bottom: 15px;
}

/* --- Подвал --- */
.footer {
    background-color: var(--color-brown);
    color: var(--color-white);
    text-align: center;
    padding: 20px 0;
    margin-top: 40px;
}

/* --- Адаптивность --- */
@media (max-width: 768px) {
    .header-content {
        flex-direction: column;
        gap: 10px;
    }
}

.hero-card h1 {

```

```

        font-size: 2.2rem;
    }

    .menu-tabs {
        overflow-x: auto;
        padding-bottom: 10px;
        justify-content: flex-start;
    }

    .tab-btn {
        white-space: nowrap;
    }
}

```

3. JavaScript (script.js)

Логика приложения: данные, рендеринг, управление корзиной.

### JavaScript

```

// --- Данные меню (Симуляция базы данных) ---
const menuItems = [
    // Стартеры
    { id: 1, name: "Начос Гранде", category: "starters", price: 450, desc: "Кукурузные чипсы с сыром, халапеньо и сальсой.", img: "https://placehold.co/300x200/FFC107/5D4037?text=Nachos" },
    { id: 2, name: "Кесадилья с курицей", category: "starters", price: 520, desc: "Тортилья с курицей, сыром и овощами.", img: "https://placehold.co/300x200/FFC107/5D4037?text=Quesadilla" },
    // Салаты
    { id: 3, name: "Салат Тако", category: "salads", price: 480, desc: "Микс салата, говядина, фасоль в хрустящей корзинке.", img: "https://placehold.co/300x200/4CAF50/FFFFFF?text=Taco+Salad" },
    { id: 4, name: "Овощной Микс", category: "salads", price: 390, desc: "Свежие овощи с лаймовой заправкой.", img: "https://placehold.co/300x200/4CAF50/FFFFFF?text=Veggie+Mix" },
    // Супы
    { id: 5, name: "Чили Кон Карне", category: "soups", price: 550, desc: "Острый суп с фаршем и фасолью.", img: "https://placehold.co/300x200/D32F2F/FFFFFF?text=Chili+Soup" },
    // Десерты
    { id: 6, name: "Чуррос", category: "desserts", price: 300, desc: "Жареное тесто с корицей и шоколадом.", img: "https://placehold.co/300x200/5D4037/FFFFFF?text=Churros" },
    // Напитки
    { id: 7, name: "Лимонад Кектус", category: "drinks", price: 250, desc: "Домашний лимонад с грушей и лаймом.", img: "https://placehold.co/300x200/81D4FA/000000?text=Lemonade" },
    { id: 8, name: "Кофе Мексикано", category: "drinks", price: 200, desc: "Черный кофе с острым перцем и шоколадом.", img: "https://placehold.co/300x200/3E2723/FFFFFF?text=Coffee" }
];

// --- Состояние приложения ---
let cart = {} // Объект для хранения выбранных товаров {id: quantity}

// --- DOM Элементы ---
const menuGrid = document.getElementById('menu-grid');
const tabButtons = document.querySelectorAll('.tab-btn');
const cartItemsContainer = document.getElementById('cart-items');
const totalAmountEl = document.getElementById('total-amount');
const cartCountEl = document.getElementById('cart-count');

```

```

// --- Инициализация ---
document.addEventListener('DOMContentLoaded', () => {
    renderMenu('all'); // Показать все блюда при загрузке
});

// --- Функции Рендеринга ---

// 1. Отображение меню
function renderMenu(category) {
    menuGrid.innerHTML = ''; // Очистка сетки

    const filteredItems = category === 'all'
        ? menuItems
        : menuItems.filter(item => item.category === category);

    filteredItems.forEach(item => {
        const itemQty = cart[item.id] || 0;

        const card = document.createElement('div');
        card.className = 'menu-item';
        card.innerHTML =
            `
            <div class="item-content">
                <h3 class="item-title">${item.name}</h3>
                <p class="item-desc">${item.desc}</p>
                <div class="item-footer">
                    <span class="item-price">${item.price} ₽</span>
                    <div class="item-controls">
                        ${itemQty > 0
                            ? `<button class="qty-btn"
                                onclick="updateQuantity(${item.id}, -1)">-</button>
                                <span>${itemQty}</span>
                                <button class="qty-btn"
                                onclick="updateQuantity(${item.id}, 1)">+</button>
                                : `<button class="btn btn-primary"
                                onclick="updateQuantity(${item.id}, 1)">В заказ</button>
                            }
                    </div>
                </div>
            `;
        menuGrid.appendChild(card);
    });
}

// 2. Обновление корзины
function renderCart() {
    cartItemsContainer.innerHTML = '';
    let total = 0;
    let totalCount = 0;
    let hasItems = false;

    // Проход по всем товарам в базе для поиска добавленных в корзину
    menuItems.forEach(item => {
        if (cart[item.id] > 0) {
            hasItems = true;
            const qty = cart[item.id];
            const sum = item.price * qty;
            total += sum;
            totalCount += qty;
        }
    });

    const cartRow = document.createElement('div');
    cartRow.className = 'cart-item';

```

```

        cartRow.innerHTML = `
            <div class="cart-item-title">${item.name}</div>
            <div class="item-controls">
                <button class="qty-btn" style="width:25px; height:25px; font-size:1rem;" onclick="updateQuantity(${item.id}, -1)">-</button>
                <span style="margin: 0 10px;">${qty}</span>
                <button class="qty-btn" style="width:25px; height:25px; font-size:1rem;" onclick="updateQuantity(${item.id}, 1)">+</button>
            </div>
            <div style="margin-left: 15px; font-weight: bold; width: 80px; text-align: right;">${sum} ₽</div>
        `;
        cartItemsContainer.appendChild(cartRow);
    }
});

if (!hasItems) {
    cartItemsContainer.innerHTML = '<p class="empty-cart-msg">Корзина пуста, амиго. Добавь немного огня!</p>';
}

totalAmountEl.textContent = total;
cartCountEl.textContent = `(${totalCount})`;
}

// --- Логика Управления ---

// Изменение количества товара
window.updateQuantity = function(id, change) {
    if (!cart[id]) cart[id] = 0;
    cart[id] += change;

    if (cart[id] <= 0) {
        delete cart[id];
    }

    // Перерисовка интерфейса
    // Находим текущую активную категорию, чтобы не сбрасывать фильтр
    const activeTab = document.querySelector('.tab-btn.active');
    const currentCategory = activeTab ? activeTab.dataset.category : 'all';

    renderMenu(currentCategory);
    renderCart();
};

// Переключение вкладок категорий
tabButtons.forEach(btn => {
    btn.addEventListener('click', () => {
        // Убираем активный класс у всех
        tabButtons.forEach(b => b.classList.remove('active'));
        // Добавляем нажатому
        btn.classList.add('active');
        // Фильтруем
        renderMenu(btn.dataset.category);
    });
});
}
);

Как запустить проект:
```

1. Создайте папку на компьютере (например, muchacho-project).
2. Внутри создайте 3 файла: index.html, style.css, script.js.
3. Скопируйте соответствующий код в каждый файл.
4. Откройте index.html в любом современном браузере.

Результат данного кода лежит в папке muchacho-project\_неудачная попытка. Данный сайт запускался в полном объёме только с компьютера, где содержались все 3 файла.

### Уточняющий промпт

При открытии файла index.html стиль и функционал пропадают, можно ли объединить все эти файлы в один html файл?

### Результат

Да, конечно! Это частая проблема, если файлы сохранены не в одной папке или браузер блокирует локальные подгрузки.

Объединение всего в один файл — отличное решение для быстрой проверки и демонстрации. Я собрал **HTML** (строку), **CSS** (стили) и **JavaScript** (функционал) в один документ.

Инструкция:

1. Создайте на компьютере файл с названием muchacho.html.
2. Скопируйте **весь** код ниже полностью.
3. Вставьте код в файл и сохраните.
4. Откройте файл в браузере.

### HTML

```
<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>MUCHACHO | Мексиканская кухня</title>
    <link
        href="https://fonts.googleapis.com/css2?family=Lobster&family=Roboto:wght@400
        ;700&display=swap" rel="stylesheet">

    <style>
        /* --- СТИЛИ (CSS) --- */
        :root {
            --color-brown: #5D4037;
            --color-white: #FFFFFF;
            --color-yellow: #FFC107;
            --color-yellow-light: #FFF8E1;
            --color-green: #2E7D32;
            --color-red: #D32F2F;
            --font-heading: 'Lobster', cursive;
            --font-body: 'Roboto', sans-serif;
        }

        * {
            box-sizing: border-box;
            margin: 0;
            padding: 0;
        }
    </style>
```

```

html {
    scroll-behavior: smooth;
}

body {
    font-family: var(--font-body);
    background-color: var(--color-yellow-light);
    color: var(--color-brown);
    line-height: 1.6;
}

.container {
    max-width: 1200px;
    margin: 0 auto;
    padding: 0 20px;
}

/* Кнопки */
.btn {
    padding: 10px 20px;
    border: none;
    border-radius: 25px;
    cursor: pointer;
    font-weight: bold;
    transition: transform 0.2s, box-shadow 0.2s;
    font-family: var(--font-body);
}

.btn:hover {
    transform: translateY(-2px);
    box-shadow: 0 4px 8px rgba(0,0,0,0.2);
}

.btn-primary {
    background-color: var(--color-brown);
    color: var(--color-white);
}

.btn-success {
    background-color: var(--color-green);
    color: var(--color-white);
    font-size: 1.2rem;
    width: 100%;
    margin-top: 10px;
}

/* Шапка */
.header {
    background-color: var(--color-brown);
    color: var(--color-yellow);
    padding: 15px 0;
    position: sticky;
    top: 0;
    z-index: 1000;
    box-shadow: 0 2px 10px rgba(0,0,0,0.3);
}

.header-content {
    display: flex;
    justify-content: space-between;
    align-items: center;
}

```

```
.logo {
    font-family: var(--font-heading);
    font-size: 2rem;
}

.nav a {
    color: var(--color-white);
    text-decoration: none;
    margin-left: 20px;
    font-weight: bold;
    transition: color 0.3s;
}

.nav a:hover {
    color: var(--color-yellow);
}

/* Hero секция */
.hero-section {
    background: linear-gradient(135deg, var(--color-yellow) 0%,
#FF9800 100%);
    padding: 80px 0;
    text-align: center;
}

.hero-card {
    background: var(--color-white);
    padding: 40px;
    border-radius: 15px;
    display: inline-block;
    box-shadow: 0 10px 20px rgba(0,0,0,0.1);
    border: 2px solid var(--color-brown);
    max-width: 500px;
    width: 100%;
}

.hero-card h1 {
    font-family: var(--font-heading);
    color: var(--color-green);
    font-size: 3rem;
    margin-bottom: 10px;
}

.auth-form {
    display: flex;
    flex-direction: column;
    gap: 15px;
    margin-top: 20px;
}

.auth-form input {
    padding: 12px;
    border: 2px solid #ddd;
    border-radius: 8px;
    font-size: 1rem;
}

/* Меню */
.menu-section {
    padding: 60px 0;
}
```

```

.section-title {
    text-align: center;
    font-family: var(--font-heading);
    font-size: 2.5rem;
    color: var(--color-brown);
    margin-bottom: 30px;
}

.menu-tabs {
    display: flex;
    justify-content: center;
    flex-wrap: wrap;
    gap: 10px;
    margin-bottom: 40px;
}

.tab-btn {
    background: transparent;
    border: 2px solid var(--color-brown);
    padding: 8px 16px;
    border-radius: 20px;
    cursor: pointer;
    font-weight: bold;
    color: var(--color-brown);
}
.tab-btn.active, .tab-btn:hover {
    background-color: var(--color-green);
    color: var(--color-white);
    border-color: var(--color-green);
}

.menu-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));
    gap: 30px;
}

.menu-item {
    background: var(--color-white);
    border-radius: 15px;
    overflow: hidden;
    box-shadow: 0 5px 15px rgba(0,0,0,0.05);
    transition: transform 0.3s;
    border-bottom: 5px solid var(--color-green);
}
.menu-item:hover {
    transform: translateY(-5px);
}

.item-img {
    width: 100%;
    height: 200px;
    background-color: #eee;
    object-fit: cover;
}

.item-content {
    padding: 20px;
}

.item-title {

```

```

        font-size: 1.25rem;
        font-weight: bold;
        margin-bottom: 5px;
    }

.item-desc {
    font-size: 0.9rem;
    color: #666;
    margin-bottom: 15px;
    min-height: 40px;
}

.item-footer {
    display: flex;
    justify-content: space-between;
    align-items: center;
}

.item-price {
    font-size: 1.2rem;
    font-weight: bold;
    color: var(--color-brown);
}

.item-controls {
    display: flex;
    align-items: center;
    gap: 10px;
}

.qty-btn {
    background: var(--color-brown);
    color: white;
    width: 30px;
    height: 30px;
    border-radius: 50%;
    border: none;
    cursor: pointer;
    font-size: 1.2rem;
    display: flex;
    justify-content: center;
    align-items: center;
}

/* Корзина */
.cart-section {
    padding: 60px 0;
    background-color: var(--color-white);
    border-top: 2px solid var(--color-brown);
}

.cart-wrapper {
    max-width: 800px;
    margin: 0 auto;
}

.cart-item {
    display: flex;
    justify-content: space-between;
    align-items: center;
    padding: 15px;
    border-bottom: 1px solid #eee;
}

```

```

.cart-item-title {
    font-weight: bold;
    flex: 1;
}

.cart-summary {
    margin-top: 30px;
    text-align: right;
    padding: 20px;
    background: var(--color-yellow-light);
    border-radius: 10px;
}

.total-price {
    font-size: 1.5rem;
    font-weight: bold;
    margin-bottom: 15px;
}

.empty-cart-msg {
    text-align: center;
    padding: 20px;
    color: #777;
    font-style: italic;
}

/* Подвал */
.footer {
    background-color: var(--color-brown);
    color: var(--color-white);
    text-align: center;
    padding: 20px 0;
    margin-top: 40px;
}

/* Адаптивность */
@media (max-width: 768px) {
    .header-content {
        flex-direction: column;
        gap: 10px;
    }
    .hero-card h1 {
        font-size: 2.2rem;
    }
    .menu-tabs {
        overflow-x: auto;
        padding-bottom: 10px;
        justify-content: flex-start;
    }
    .tab-btn {
        white-space: nowrap;
    }
}
</style>
</head>
<body>

<header class="header">
    <div class="container header-content">
        <div class="logo">MUCHACHO <img alt="Muchacho logo" /></div>
        <nav class="nav">
            <a href="#auth">Войти</a>

```

```

        <a href="#menu">Меню</a>
        <a href="#cart">Корзина <span id="cart-count">(0)</span></a>
    </nav>
</div>
</header>

<section id="auth" class="hero-section">
    <div class="container">
        <div class="hero-card">
            <h1>Hola, Amigo!</h1>
            <p>Добро пожаловать в Muchacho. Заказывай быстро, ешь
вкусно!</p>
            <form class="auth-form" onsubmit="event.preventDefault();
alert('Вы авторизованы!');>
document.getElementById('menu').scrollIntoView();">
                <input type="text" placeholder="Ваше имя" required>
                <input type="number" placeholder="Номер столика" min="1"
max="20" required>
                <button type="submit" class="btn btn-primary">Начать
заказ</button>
            </form>
        </div>
    </div>
</section>

<section id="menu" class="menu-section">
    <div class="container">
        <h2 class="section-title">Наше Меню</h2>

        <div class="menu-tabs">
            <button class="tab-btn active" data-
category="all">Все</button>
            <button class="tab-btn" data-
category="starters">Стarterы</button>
            <button class="tab-btn" data-
category="salads">Салаты</button>
            <button class="tab-btn" data-category="soups">Супы</button>
            <button class="tab-btn" data-
category="desserts">Десерты</button>
            <button class="tab-btn" data-
category="drinks">Напитки</button>
        </div>

        <div id="menu-grid" class="menu-grid">
        </div>
    </div>
</section>

<section id="cart" class="cart-section">
    <div class="container">
        <h2 class="section-title">Ваш Заказ</h2>
        <div class="cart-wrapper">
            <div id="cart-items" class="cart-items">
                <p class="empty-cart-msg">Корзина пуста, амиго. Добавь
немного огня!</p>
            </div>
            <div class="cart-summary">
                <div class="total-price">Итого: <span id="total-
amount">0</span> ₽</div>
                <button class="btn btn-success" onclick="alert('Заказ
отправлен на кухню! Ожидайте.')">Оформить заказ</button>
            </div>
        </div>
    </div>
</section>

```

```

        </div>
    </section>

<footer class="footer">
    <div class="container">
        <p>&copy; 2023 Restaurant Muchacho. Buen provecho!</p>
    </div>
</footer>

<script>
    // --- Данные меню ---
    const menuItems = [
        { id: 1, name: "Начос Гранде", category: "starters", price: 450,
        desc: "Кукурузные чипсы с сыром, халапеньо и сальсой.", img:
        "https://placehold.co/300x200/FFC107/5D4037?text=Nachos" },
        { id: 2, name: "Кесадилья с курицей", category: "starters",
        price: 520, desc: "Тортилья с курицей, сыром и овощами.", img:
        "https://placehold.co/300x200/FFC107/5D4037?text=Quesadilla" },
        { id: 3, name: "Салат Тако", category: "salads", price: 480,
        desc: "Микс салата, говядина, фасоль в хрустящей корзинке.", img:
        "https://placehold.co/300x200/4CAF50/FFFFFF?text=Taco+Salad" },
        { id: 4, name: "Овощной Микс", category: "salads", price: 390,
        desc: "Свежие овощи с лаймовой заправкой.", img:
        "https://placehold.co/300x200/4CAF50/FFFFFF?text=Veggie+Mix" },
        { id: 5, name: "Чили Кон Карне", category: "soups", price: 550,
        desc: "Острый суп с фаршем и фасолью.", img:
        "https://placehold.co/300x200/D32F2F/FFFFFF?text=Chili+Soup" },
        { id: 6, name: "Чуррос", category: "desserts", price: 300, desc:
        "Жареное тесто с корицей и шоколадом.", img:
        "https://placehold.co/300x200/5D4037/FFFFFF?text=Churros" },
        { id: 7, name: "Лимонад Кактус", category: "drinks", price: 250,
        desc: "Домашний лимонад с грушей и лаймом.", img:
        "https://placehold.co/300x200/81D4FA/000000?text=Lemonade" },
        { id: 8, name: "Кофе Мексикано", category: "drinks", price: 200,
        desc: "Черный кофе с острым перцем и шоколадом.", img:
        "https://placehold.co/300x200/3E2723/FFFFFF?text=Coffee" }
    ];

    // --- Состояние приложения ---
    let cart = {};

    // --- DOM Элементы ---
    const menuGrid = document.getElementById('menu-grid');
    const tabButtons = document.querySelectorAll('.tab-btn');
    const cartItemsContainer = document.getElementById('cart-items');
    const totalAmountEl = document.getElementById('total-amount');
    const cartCountEl = document.getElementById('cart-count');

    // --- Инициализация ---
    document.addEventListener('DOMContentLoaded', () => {
        renderMenu('all');
    });

    // --- Функции Рендеринга ---
    function renderMenu(category) {
        menuGrid.innerHTML = '';

        const filteredItems = category === 'all'
            ? menuItems
            : menuItems.filter(item => item.category === category);

        filteredItems.forEach(item => {
            const itemQty = cart[item.id] || 0;

```

```

        const card = document.createElement('div');
        card.className = 'menu-item';
        card.innerHTML =
            `
            <div class="item-content">
                <h3 class="item-title">${item.name}</h3>
                <p class="item-desc">${item.desc}</p>
                <div class="item-footer">
                    <span class="item-price">${item.price} ₽</span>
                    <div class="item-controls">
                        ${itemQty > 0
                            ? `<button class="qty-btn"
                                onclick="updateQuantity(${item.id}, -1)">-</button>
                                <span>${itemQty}</span>
                                <button class="qty-btn"
                                onclick="updateQuantity(${item.id}, 1)">+</button>
                                : `<button class="btn btn-primary"
                                onclick="updateQuantity(${item.id}, 1)">B үзүүлэх</button>
                            }
                        </div>
                    </div>
                </div>
            `;
        menuGrid.appendChild(card);
    });
}

function renderCart() {
    cartItemsContainer.innerHTML = '';
    let total = 0;
    let totalCount = 0;
    let hasItems = false;

    menuItems.forEach(item => {
        if (cart[item.id] > 0) {
            hasItems = true;
            const qty = cart[item.id];
            const sum = item.price * qty;
            total += sum;
            totalCount += qty;

            const cartRow = document.createElement('div');
            cartRow.className = 'cart-item';
            cartRow.innerHTML =
                `<div class="cart-item-title">${item.name}</div>
                <div class="item-controls">
                    <button class="qty-btn" style="width:25px;
height:25px; font-size:1rem;" onclick="updateQuantity(${item.id}, -1)">-
                    <span style="margin: 0 10px;">${qty}</span>
                    <button class="qty-btn" style="width:25px;
height:25px; font-size:1rem;" onclick="updateQuantity(${item.id},
1)">+</button>
                </div>
                <div style="margin-left: 15px; font-weight: bold;
width: 80px; text-align: right;">${sum} ₽</div>
            `;
            cartItemsContainer.appendChild(cartRow);
        }
    });
}

```

```

        if (!hasItems) {
            cartItemsContainer.innerHTML = '<p class="empty-cart-' +
msg">Корзина пуста, амigo. Добавь немного огня!</p>';
        }

        totalAmountEl.textContent = total;
        cartCountEl.textContent = `(${totalCount})`;
    }

    // --- Логика Управления ---
    window.updateQuantity = function(id, change) {
        if (!cart[id]) cart[id] = 0;
        cart[id] += change;

        if (cart[id] <= 0) {
            delete cart[id];
        }

        const activeTab = document.querySelector('.tab-btn.active');
        const currentCategory = activeTab ? activeTab.dataset.category :
'all';

        renderMenu(currentCategory);
        renderCart();
    };

    tabButtons.forEach(btn => {
        btn.addEventListener('click', () => {
            tabButtons.forEach(b => b.classList.remove('active'));
            btn.classList.add('active');
            renderMenu(btn.dataset.category);
        });
    });

```

</script>

</body>

</html>

## Проверка результатов по критериям

- Лендинг содержит все запрошенные блоки — Да;
- Код валиден и не содержит синтаксических ошибок — Да;
- Страница корректно отображается в браузере — Да;
- Реализована адаптивная вёрстка — Да;
- Код легко читаем и подходит для повторного использования — Да.

По шкале от 1 до 5:

- соответствие дизайна и содержания исходному описанию проекта — 5;
- техническое качество клиентского кода — 5;
- общую пригодность лендинга для практического применения — 5.

## **ВЫВОДЫ**

В ходе работы было проанализировано умение Gemini 3 создавать одностраничный сайт по детализированному промпту. Результат удивил, с учетом уточняющего промпта и неудачной попытки, создание одностраничного сайта заняло 5 минут. С возможностью открывать с телефона и повторным использованием.