**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ**

**КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»**

Факультет «Информационные системы и технологии»

Кафедра «Интеллектуальные системы автоматизации и управления»

Направление подготовки: 09.03.02. Информационные системы и технологии

Направленность (профиль): Системное и прикладное программирование информационных систем

**КУРСОВОЙ ПРОЕКТ**

по дисциплине «Основы интернет-технологий»

на тему:

«Разработка web-сайта «Not to forget» с

использованием технологий HTML, CSS и JavaScript».

|  |  |
| --- | --- |
| Выполнил студент группы | ИСТ-461 |
| Димова Анастасия Дмитриевна  *Фамилия И.О.* | |

|  |  |  |
| --- | --- | --- |
| *оценка* | Руководитель | к.т.н., доцент каф. САР  *уч.степень, уч.звание* |
| *дата,подпись* | Любимов Александр Геннадьевич  *Фамилия И.О.* | |

Члены комиссии:

|  |  |
| --- | --- |
| *дата,подпись* | *Фамилия И.О.* |
| *дата,подпись* | *Фамилия И.О.* |

Санкт-Петербург



2025



**ЗАДАНИЕ**

I. Тема работы: Разработка статического web-сайта с использованием

технологий HTML, CSS и JavaScript.

II. Задание на работу:

Разработать сайт со следующими требованиями:

1. Тема сайта - произвольная (по согласованию с преподавателем).

2. Объем сайта – 3-5 web-страниц.

3. Обязательные требования по применению технологий HTML, CSS и JavaScript при разработке сайта:

3.1. HTML:

• Списки (нумерованные, маркированные, определений);

• Таблицы (с текстовым и графическим контентом; таблицы,

вложенные в таблицу);

• Ссылки (внешние, внутренние, якорь)

3.2. CSS:

• Стиль, встроенный в тег;

• Внутренняя таблица стилей;

• Связывание с внешней таблицей стилей;

• Селекторы (элемента, класса, ID селектор).

3.3. Работа с JavaScript:

• Способы встраивания JavaScript-сценария в HTML-документ:

- с помощью тега script;

- с помощью обработчика событий;

- связывание с внешним файлом сценария.

• Методы ввода/вывода данных:

- Метод prompt ( );

- Метод alert ( ).

Содержание

**Введение**

В современном мире с его высоким темпом жизни и постоянным потоком информации многие сталкиваются с проблемой организации личного времени. Невыполненные задачи, забытые дела и отсутствие системного подхода к формированию полезных привычек снижают продуктивность и увеличивают уровень стресса. В связи с этим особую актуальность приобретают инструменты, помогающие эффективно планировать время, отслеживать выполнение задач и постепенно внедрять новые привычки.

Разработка статического веб-приложения для учета дел и формирования привычек представляет собой практическое решение этой проблемы. Такой сайт не требует серверной части, что делает его простым в развертывании и использовании, но при этом сохраняет функциональность, достаточную для базового планирования. Приложение включает в себя:

* **трекер привычек** – для регулярного отслеживания прогресса;
* **список задач** – для фиксации текущих дел;
* **раздел с полезной информацией** – для обучения пользователей эффективным методикам планирования.

**Целью** является разработка статического веб-приложения, помогающего пользователям систематизировать повседневные задачи и формировать полезные привычки.

**Задачи**:

1. Провести анализ существующих решений в области трекеров привычек и планировщиков задач.
2. Разработать удобный и интуитивно понятный пользовательский интерфейс в спокойных светлых тонах. Использовать голубые и серые оттенки, белый цвет. Сделать макет в figma для чёткого представления будущего внешнего вида сайта.
3. Реализовать функционал добавления и удаления задач и привычек.
4. Обеспечить удобство пользования путём добавления на страницу календаря, отметки сегодняшнего дня недели, регистрации по электронной почте и паролю, авторизации.
5. Добавить раздел с рекомендациями по эффективному планированию.

**Используемые методы верстки и дизайна**

При создании сайта применялись следующие технологии и подходы:

**1. Верстка и структура**

**•** HTML5 – для семантической разметки страницы.

• CSS3 – для стилизации элементов.

**2. Дизайн и интерфейс**

Минималистичный дизайн

* Чистая, ненагруженная композиция с акцентом на функциональность.

Интерактивность через JavaScript

* Динамическое добавление и удаление задач и привычек.
* Сохранение данных локально в рамках одного сеанса.
* Обработка событий – клики, наведение, ввод текста.
* Валидация форм – проверка корректности вводимых данных.

Визуальная обратная связь

* Подсветка дней формирования привычки.
* Анимации кнопок (изменение размера или цвета при наведении).

**3. Инструменты разработки**

VS Code

* Использование расширений для ускорения разработки:
  + Live Server – мгновенный просмотр изменений;
  + Prettier – автоматическое форматирование кода;

Git и GitHub

* Система контроля версий для отслеживания изменений.

(ссылка на проект: https://github.com/pryfmse/coursework\_web)

Google Fonts

* Подключены шрифты:
  + Yanone Kaffeesatz – для основного текста (хорошая читаемость);
  + Pacifico – для заголовков (придает стильность).

Figma

* Создание прототипа интерфейса перед версткой.
* Определение цветовой палитры, шрифтов и расположения элементов.

Разрабатываемое веб-приложение сочетает в себе простоту статического сайта с функциональностью, достаточной для базового управления задачами и привычками. В дальнейшем проект может быть расширен за счет добавления статистики, синхронизации между устройствами и хранения данных на удаленном сервере.

**Основная часть**

Веб-разработка является ключевой областью современных информационных технологий. Статические сайты, построенные на базе HTML, CSS и JavaScript, остаются основой для создания интерактивных и визуально привлекательных веб-ресурсов. В данной работе рассматриваются теоретические аспекты применения этих технологий, а также их практическая реализация в рамках разработки сайта для планирования задач и привычек.

### **Глава 1. Основные средства HTML, CSS и JavaScript для создания статических веб-страниц**

#### **1.1 Назначение и основные средства языка HTML**

**Назначение и история развития:**  
HTML (HyperText Markup Language) — это язык разметки, созданный для структурирования контента веб-страниц. Разработан Тимом Бернерсом-Ли в 1991 году как инструмент для обмена научными документами. Сейчас HTML эволюционировал до версии HTML5 (2014), которая добавила семантические теги, мультимедийную поддержку и API для интеграции с JavaScript.

**Ключевые особенности:**

1. **Теги и атрибуты**:
   * Теги определяют элементы страницы:
     + <h1>–<h6> — заголовки;
     + <p> — абзацы;
     + <a href="..."> — гиперссылки.
   * Атрибуты задают свойства элементов:
     + class и id для стилизации и идентификации;
     + src для указания источника изображений (<img>);
2. **Семантическая разметка**:  
   HTML5 ввел теги, улучшающие понимание структуры:
   * <header> — шапка страницы;
   * <nav> — навигационное меню;
   * <article> — независимый контент (статья);
   * <footer> — нижняя часть.  
     Семантика повышает доступность для скринридеров и SEO-оптимизацию.
3. **Формы и интерактивность**:  
   Формы (<form>) собирают данные пользователя. Элементы:
   * <input type="text"> — текстовое поле;
   * <input type="email"> — валидация email;
   * <button> — кнопки действий.

**Пример структуры HTML-документа:**

1. <!DOCTYPE html>
2. <html lang="ru">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <title>Семантическая страница</title>
7. </head>
8. <body>
9. <header>
10. <h1>Заголовок сайта</h1>
11. <nav>
12. <a href="#about">О нас</a>
13. <a href="#contacts">Контакты</a>
14. </nav>
15. </header>
16. <main>
17. <article>
18. <h2>Статья</h2>
19. <p>Текст статьи...</p>
20. </article>
21. </main>
22. <footer>
23. <p>© 2023</p>
24. </footer>
25. </body>
26. </html>

#### **1.2 Назначение и возможности каскадных таблиц стилей CSS**

**Назначение и принципы работы:**  
CSS (Cascading Style Sheets) — язык стилей, отделяющий оформление от структуры HTML. Позволяет управлять макетом, цветами, шрифтами и адаптивностью. Основные принципы:

* **Каскадирование**: Приоритет стилей определяется специфичностью селекторов и порядком их объявления.
* **Наследование**: Дочерние элементы наследуют стили родителя (например, font-family).

**Основные возможности:**

1. **Селекторы**:
   * **Базовые**:
     + тег (например, div);
     + .class (например, .container);
     + #id (например, #header).
   * **Комбинированные**:
     + div.class — элемент с классом;
     + parent > child — прямой потомок.
2. **Свойства стилей**:
   * **Текст**:
     + font-size, font-family, text-align;
   * **Блоки**:
     + margin, padding, border, box-shadow;
   * **Позиционирование**:
     + position: absolute, fixed, relative;
     + display: flex, grid, block.
3. **Адаптивный дизайн**:  
   Медиа-запросы (@media) меняют стили в зависимости от устройства:
4. @media (max-width: 768px) {
5. body {
6. font-size: 14px;
7. }
8. }
9. **Flexbox и Grid**:
   * **Flexbox** — одномерное выравнивание элементов (горизонтально или вертикально):
10. .container {
11. display: flex;
12. justify-content: center;
13. }
    * **Grid** — двумерная сетка для сложных макетов:
14. .grid-container {
15. display: grid;
16. grid-template-columns: repeat(3, 1fr);
17. }
18. **Анимации и переходы**:
    * transition — плавное изменение свойств:
19. button {
20. transition: background-color 0.3s ease;
21. }
    * @keyframes — создание кадров анимации:
22. @keyframes fadeIn {
23. from { opacity: 0; }
24. to { opacity: 1; }
25. }

**Пример стилизации:**

1. /\* Глобальные стили \*/
2. body {
3. font-family: Arial, sans-serif;
4. margin: 0;
5. }
6. /\* Стилизация навигации \*/
7. nav {
8. background-color: #333;
9. padding: 1rem;
10. }
11. nav a {
12. color: white;
13. text-decoration: none;
14. margin-right: 20px;
15. }
16. /\* Адаптивность \*/
17. @media (max-width: 480px) {
18. nav {
19. padding: 0.5rem;
20. }
21. }

#### **1.3 Технологии создания JavaScript-сценариев в веб-страницах**

**Назначение и роль в веб-разработке:**  
JavaScript (JS) — язык программирования, добавляющий интерактивность: обработка событий, динамическое изменение контента, работа с API.

**Основные концепции:**

* **Интеграция с HTML**:
  + Встроенные скрипты через тег <script>:

1. <script>
2. alert("Hello, World!");
3. </script>
   * Подключение внешних файлов:
4. <script src="script.js"></script>

* **События**:  
  JS реагирует на действия пользователя:
  + click — клик мышью;
  + mouseover — наведение курсора;
  + submit — отправка формы.  
    Пример обработки события:

1. document.querySelector("button").addEventListener("click", function() {
2. alert("Кнопка нажата!");
3. });

**Пример JS-кода:**

1. // Динамическое создание элемента
2. const newElement = document.createElement("div");
3. newElement.textContent = "Новый блок";
4. document.body.appendChild(newElement);
5. // Анимация при прокрутке
6. window.addEventListener("scroll", function() {
7. if (window.scrollY > 100) {
8. document.querySelector("header").style.backgroundColor = "#555";
9. }
10. });

**Глава 2. Описание технологий разработанного web-сайта**

**2.1 Структура сайта**

Сайт состоит из трёх статических страниц, связанных навигацией:

1. **Главная страница (index.html)**
   * Содержит блоки для управления задачами и привычками, календарь.
   * Элементы:
     + Приветствие пользователя.
     + Кнопки добавления задач и привычек.
     + Календарь с текущим месяцем.
     + Кнопки перехода к страницам регистрации/входа и странице о планировании с теоретической информацией.
2. **Страница входа/регистрации (enter.html)**
   * Формы для авторизации и регистрации пользователя.
   * Анимация переключения между формами.
3. **Страница о планировании (about\_planning.html)**
   * Теоретическая информация о методах формирования привычек и планирования задач.
   * Использование таблиц, списков и определений.

**Навигация:**

* Между страницами переход осуществляется через гиперссылки (<a href="...">).
* На главной странице кнопка «О планировании» перенаправляет на about\_planning.html.

**Пример структуры HTML (index.html):**

1. <header>
2. <h1><span id="greeting"></span>, <span id="name"></span></h1>
3. <button onclick="logout()">профиль</button>
4. </header>
5. <main>
6. <table>
7. <tr>
8. <td>
9. <section>
10. <div class="gor">
11. <h2>Задачи</h2>
12. <button onclick="add\_task()">+</button>
13. </div>
14. </section>
15. </td>
16. <td>
17. <div class="calendar">...</div>
18. </td>
19. </tr>
20. </table>
21. </main>

**2.2 Описание технологий HTML**

**Основные элементы:**

1. **Семантические теги:**
   * <header>, <nav>, <main>, <section>, <footer> — для структурирования контента.
   * <article>, <table>, <dl> — для представления информации.
2. **Формы:**
   * Вход и регистрация (enter.html):
3. <form class="form">
4. <input type="email" placeholder="Email">
5. <input type="password" placeholder="Password">
6. <button onclick="Sign\_in(event)">Войти</button>
7. </form>
   * Использование атрибутов: required, type, placeholder.
8. **Таблицы и списки:**
   * Таблица методов планирования (about\_planning.html):
9. <table>
10. <thead>
11. <tr>
12. <th>Метод</th>
13. <th>Описание</th>
14. </tr>
15. </thead>
16. <tbody>
17. <tr>
18. <td>Матрица Эйзенхауэра</td>
19. <td>
20. <ul>
21. <li>Важно + Срочно</li>
22. </ul>
23. </td>
24. </tr>
25. </tbody>
26. </table>

**2.3 Описание стилей CSS**

**Ключевые аспекты:**

1. **Модульность:**
   * Стили разделены на файлы: style\_main.css, style\_enter.css, style\_planning.css.
2. **Адаптивность:**
   * Использование Flexbox для выравнивания элементов:
3. .gor {
4. display: flex;
5. align-items: center;
6. }
7. **Анимации:**
   * Эффект переключения форм входа/регистрации:
8. #chk:checked ~ .register {
9. transform: translateY(-60%);
10. transition: 0.8s ease-in-out;
11. }
12. **Стилизация состояний:**
    * Изменение цвета дней недели для привычек:
13. .h4-green {
14. color: #27ae60;
15. }
16. .h4-red {
17. color: #e67e22;
18. }

**Пример медиа-запроса:**

1. @media (max-width: 768px) {
2. nav ul {
3. flex-direction: column;
4. }
5. }

**2.4 Описание JavaScript-сценариев**

**Основные функции:**

1. **Работа с формами:**
   * Регистрация и авторизация:
2. function Sign\_up(event) {
3. const email = document.getElementById("email\_reg").value;
4. registered[email] = [password, username];
5. alert("Успешная регистрация");
6. }
7. **Динамическое добавление элементов:**
   * Создание новой задачи:
8. function add\_task() {
9. const task = prompt("Введите задачу:");
10. const newTask = document.createElement("div");
11. newTask.innerHTML = `<div class="block">${task}</div>`;
12. div.append(newTask);
13. }
14. **Работа с DOM:**
    * Обновление приветствия в зависимости от времени:
15. if (hours < 12) {
16. document.getElementById("greeting").textContent = "Доброе утро";
17. }
18. **Календарь:**
    * Генерация дней месяца:
19. for (let day = 1; day <= daysInMonth; day++) {
20. const dayCell = document.createElement("div");
21. dayCell.className = "day";
22. dayCell.innerText = day;
23. daysContainer.appendChild(dayCell);
24. }

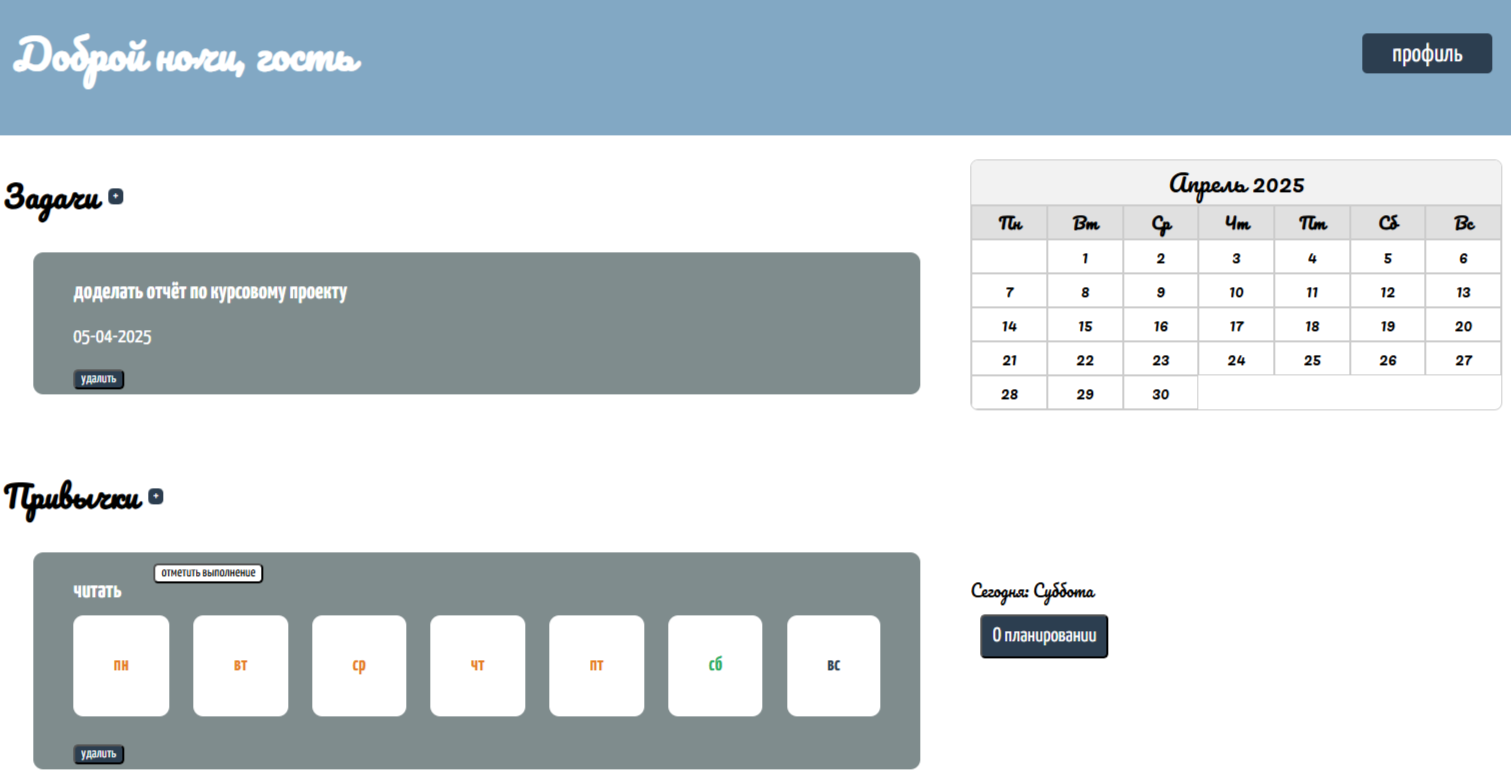
**Глава 3. Описание интерфейса и принципа работы web-сайта**

**Глава 3. Описание интерфейса и принципа работы web-сайта**

**3.1 Описание интерфейса web-сайта**

Интерфейс сайта состоит из трёх взаимосвязанных страниц, каждая из которых решает определённые задачи:

1. **Главная страница (index.html)**

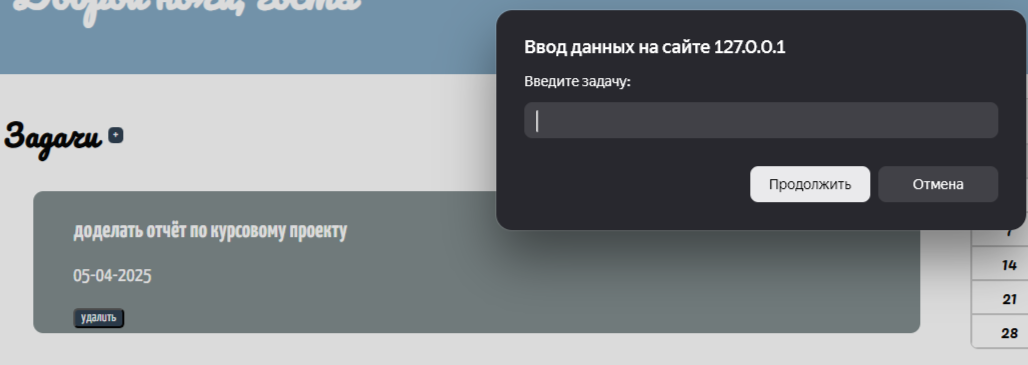


* + **Элементы:**
    - **Шапка (<header>):**



* + - * Приветствие пользователя с динамическим текстом (например, "Добрый день, Гость").
      * Кнопка "Профиль" для входа в аккаунт.

1. <header>
2. <h1><span id="greeting"></span>, <span id="name"></span></h1>
3. <button class="button-header" onclick="logout()">профиль</button>
4. </header>
   * + **Блок задач:**



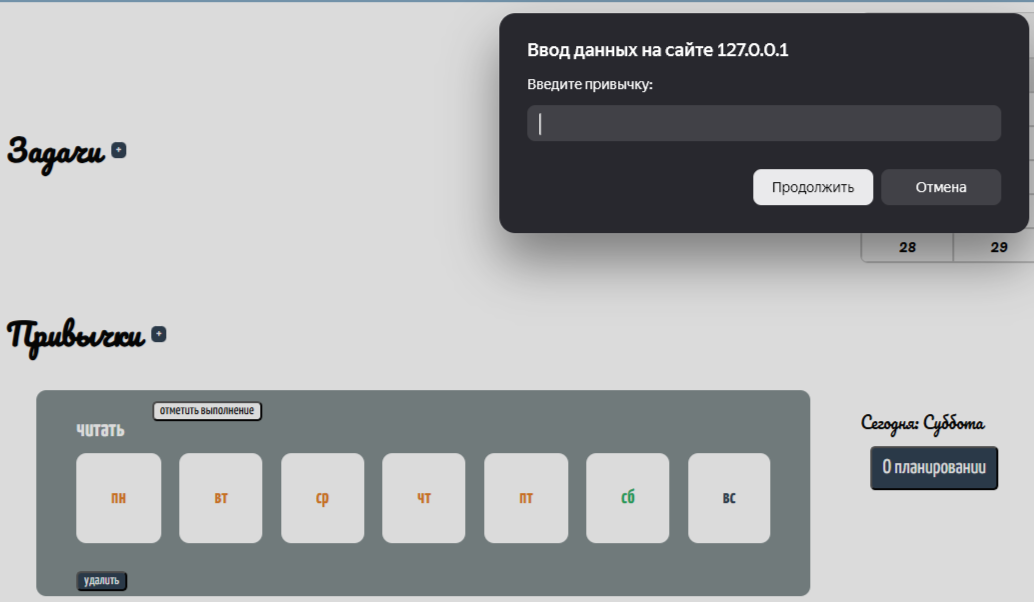
* + - * Заголовок "Задачи" с кнопкой добавления новых задач (+).
      * Динамический список задач, создаваемых пользователем.

1. <div class="gor">
2. <h2>Задачи</h2>
3. <button class="button-add" onclick="add\_task()">+</button>
4. </div>
5. <div class="list"></div>
   * + **Календарь:**



* + - * Отображение текущего месяца и дней.

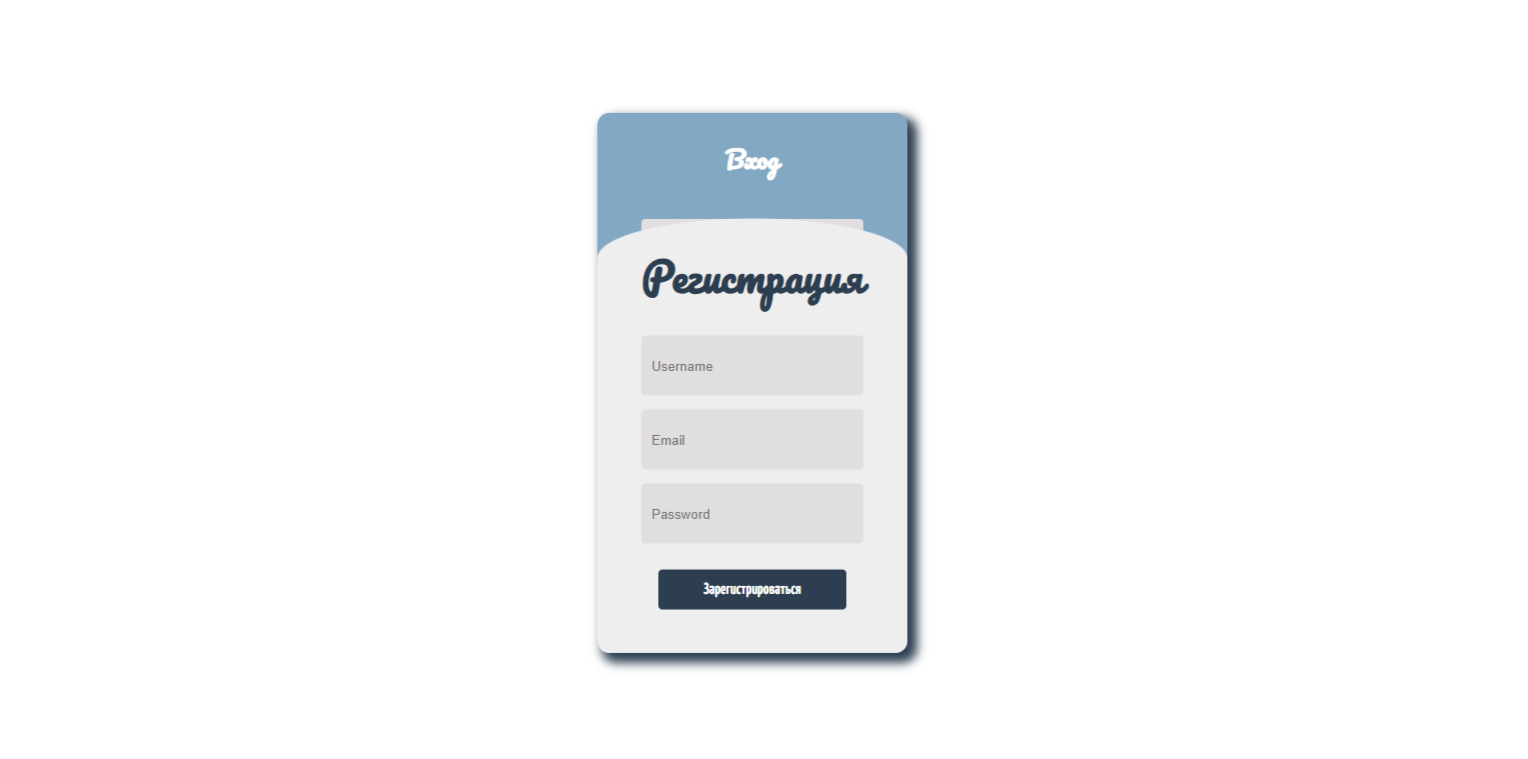
1. <div class="calendar">
2. <div class="month" id="month"></div>
3. <div class="days" id="days"></div>
4. </div>
   * + **Блок привычек:**



* + - * Заголовок "Привычки" с кнопкой добавления.
      * Визуализация дней недели с цветовой индикацией выполнения.

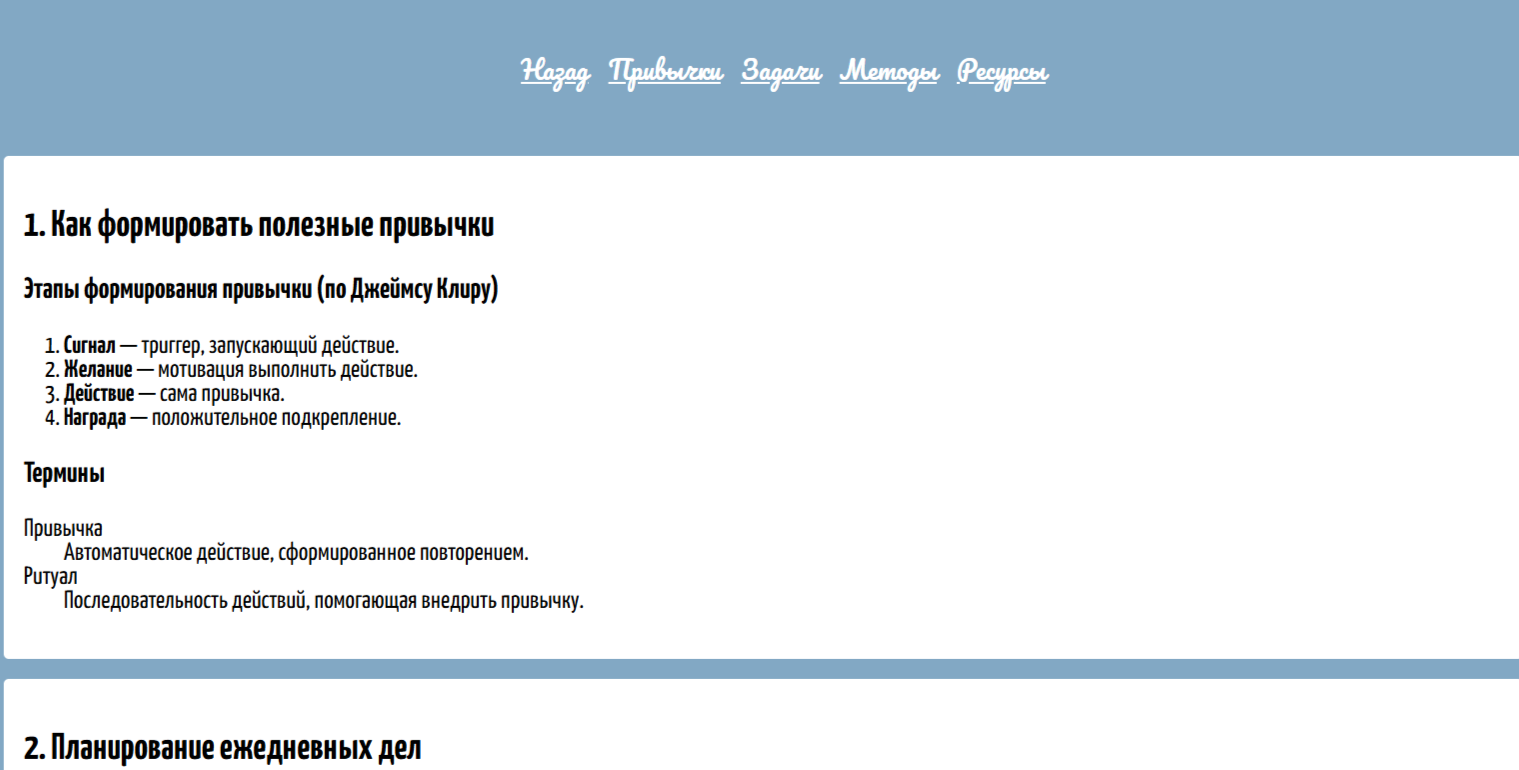
1. <div class="gor">
2. <h2>Привычки</h2>
3. <button class="button-add" onclick="add\_habit()">+</button>
4. </div>
5. **Страница входа/регистрации (enter.html)**





* + **Элементы:**
    - Переключаемая форма между входом и регистрацией.
    - Анимация сдвига форм при переключении.

1. <div class="main">
2. <input type="checkbox" id="chk">
3. <div class="login">
4. <form class="form">
5. <label for="chk">Вход</label>
6. <input type="email" placeholder="Email">
7. <input type="password" placeholder="Password">
8. </form>
9. </div>
10. <div class="register">
11. <form class="form">
12. <label for="chk">Регистрация</label>
13. <input type="text" placeholder="Username">
14. </form>
15. </div>
16. </div>
17. **Страница о планировании (about\_planning.html)**



* + **Элементы:**
    - Навигационное меню с ссылками на разделы.
    - Таблицы, списки и определения для структурирования информации.

1. <section id="habits">
2. <h2>Как формировать полезные привычки</h2>
3. <ol>
4. <li><strong>Сигнал</strong> — триггер, запускающий действие.</li>
5. </ol>
6. </section>

**3.2 Описание принципа работы web-сайта**

**Взаимодействие технологий:**

1. **HTML** — создаёт структуру страниц:
   * Семантические теги (<section>, <article>) улучшают доступность.
   * Формы (<form>) собирают данные пользователя.
2. **CSS** — обеспечивает визуальное оформление:
   * **Адаптивность** через медиа-запросы:
3. @media (max-width: 768px) {
4. nav ul { flex-direction: column; }
5. }
   * **Анимации** для переключения форм:
6. #chk:checked ~ .register {
7. transform: translateY(-60%);
8. transition: 0.8s ease-in-out;
9. }
10. **JavaScript** — добавляет интерактивность:
    * **Динамическое создание задач:**
11. function add\_task() {
12. const task = prompt("Введите задачу:");
13. const newTask = document.createElement("div");
14. newTask.innerHTML = `<div class="block">${task}</div>`;
15. document.querySelector(".list").append(newTask);
16. }
    * **Работа с датой и временем:**
17. const date = new Date();
18. document.getElementById("day").textContent = daysOfWeek[date.getDay()];
    * **Валидация форм:**
19. function Sign\_in(event) {
20. event.preventDefault();
21. if (!email.value.includes("@")) alert("Некорректный email!");
22. }

**Пример взаимодействия:**

* При нажатии кнопки "Отметить выполнение" в блоке привычек, JavaScript изменяет цвет текущего дня:

1. const Notice = (event) => {
2. const todayElement = event.target.closest(".block").querySelector(".h4-now");
3. todayElement.classList.replace("h4-now", "h4-green");
4. };

**Заключение**

**1. Использованные технологии**  
В ходе разработки статического web-сайта были применены следующие технологии:

**1. HTML5**

* **Семантическая верстка**:  
  Использованы теги <header>, <main>, <section>, <nav>, <footer> для создания логической структуры страниц, что улучшает SEO и доступность.

1. <header>...</header>
2. <section id="habits">...</section>

* **Формы**:  
  Реализованы интерактивные формы для регистрации, авторизации и добавления задач:

1. <form class="form">
2. <input type="email" placeholder="Email">
3. <button type="submit">Войти</button>
4. </form>

**2. CSS3**

* **Flexbox и Grid**:
  + **Flexbox** для горизонтального выравнивания элементов (навигация, блоки задач):

1. .gor {
2. display: flex;
3. justify-content: space-between;
4. }
   * **Grid** для создания календаря:
5. .days {
6. display: grid;
7. grid-template-columns: repeat(7, 1fr);
8. }

* **Анимации**:
  + Плавные переходы для кнопок и форм:

1. .button-header {
2. transition: opacity 0.3s;
3. }
   * Анимация "свечения" кнопки профиля через @keyframes:
4. @keyframes anim {
5. 0% { transform: scale(0) rotate(45deg); opacity: 0; }
6. 100% { transform: scale(50) rotate(45deg); opacity: 0; }
7. }

* **Кастомные стили**:
  + Стилизация состояний дней привычек (.h4-now, .h4-future);
  + Тени и скругления для визуальной глубины:

1. .block {
2. box-shadow: 0 2px 10px rgba(0,0,0,0.1);
3. border-radius: 10px;
4. }

**3. JavaScript (ES6+)**

* **Работа с DOM**:  
  Динамическое создание и удаление элементов (задачи, привычки):

1. function add\_task() {
2. const newTask = document.createElement("div");
3. newTask.innerHTML = `<div class="block">${task}</div>`;
4. div.append(newTask);
5. }

* **Обработка событий**:
  + Клики по кнопкам: onclick="Sign\_in(event)";
* **Локальное хранилище**:

1. let registered = {}; // Имитация базы данных
2. registered[email] = [password, username];

* **Работа с датами**:
  + Определение текущего дня недели и месяца:

1. const dayIndex = new Date().getDay();
2. const monthNames = ["Январь", "Февраль", ...];
   * Динамическая генерация календаря.

**4. Вспомогательные инструменты**

* **Google Fonts**:  
  Подключены шрифты для улучшения визуального восприятия:

1. <link href="https://fonts.googleapis.com/css2?family=Pacifico&display=swap" rel="stylesheet">

* **Иконки**:  
  Использованы эмодзи для интуитивной навигации

**5. Архитектура проекта**

* **Модульность**:  
  Разделение кода на файлы:
  + index.html — главная страница;
  + style\_index.css — стили основной страницы;
  + script.js — логика взаимодействия.
* **Кроссбраузерность**:  
  Проверка корректной работы в современных браузерах (Chrome, Firefox, Edge).

**6. Особенности реализации**

* **Интерактивность**:
  + Переключение между формами входа и регистрации через CSS-анимации;
  + Динамическое изменение цвета дней привычек (красный/зеленый).
* **Валидация**:
  + Проверка формата даты с помощью регулярных выражений:

1. /^\d{2}-\d{2}-\d{4}$/.test(date)

**Итог**: Проект демонстрирует комплексное применение современных веб-технологий. Каждый инструмент был использован целенаправленно:

* HTML — для структуры,
* CSS — для дизайна и адаптивности,
* JavaScript — для логики и интерактивности.

**2. Степень выполнения задания**  
Курсовая работа выполнена в полном объеме согласно задуманному:

* Реализована система регистрации и авторизации пользователей (клиентская часть);
* Создан интерактивный календарь с динамической генерацией дней месяца;
* Разработан трекер привычек с визуализацией прогресса (цветовая индикация дней);
* Реализован менеджер задач с возможностью добавления, удаления и отслеживания дедлайнов;
* Интегрированы методы планирования (матрица Эйзенхауэра, правило 1-3-5);
* Добавлены анимации (кнопки, переходы между формами);

**3. Предложения и выводы**  
**Выводы:**

* Реализованный функционал соответствует целям курсовой работы — создание инструмента для личного планирования;
* Особое внимание уделено интерактивности и визуальной составляющей, что повышает удобство использования.

**Предложения для дальнейшего развития:**

* Внедрение бэкенда (например, на Node.js или Python) для хранения данных пользователей;
* Добавление напоминаний (интеграция с Email/Telegram API);
* Расширение функционала календаря (синхронизация с Google Calendar);
* Реализация статистики привычек и задач (графики на Chart.js);
* Улучшение мобильной версии;
* Внедрение PWA-функционала для оффлайн-доступа.

**Итог**: Сайт готов к использованию в качестве личного планировщика, а его архитектура позволяет легко масштабировать функционал.

**Список используемой литературы**

1. **Бородин, М. В.** Интернет-технологии: учеб. пособие / М. В. Бородин, Е. А. Титенко. — Курск: Юго-Зап. гос. ун-т, 2013. — 140 с.
2. **Квинт, И. И.** HTML, XHTML и CSS на 100 % / И. И. Квинт. — СПб.: Питер, 2010. — 384 с.
3. **Крокфорд, Д.** JavaScript: сильные стороны / Д. Крокфорд; пер. с англ. А. А. Слинкина. — СПб.: Питер, 2012. — 176 с.
4. **Эндрю, Р.** CSS: 100 и 1 совет / Р. Эндрю; пер. с англ. И. В. Иванова. — 3-е изд. — СПб.: Символ-Плюс, 2010. — 336 с.
5. **MDN Web Docs [Электронный ресурс] / Mozilla Foundation. —** Режим доступа: https://developer.mozilla.org/ru/

**Приложение**

**Index.html**

1. <!DOCTYPE html>
2. <html lang="ru">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <title>Not to forget</title>
8. <!-- Стили для горизонтального выравнивания -->
9. <style>
10. .gor {
11. display: flex; /\* расположение элементов по горизонтали \*/
12. align-items: center; /\* элементы по вертикали \*/
13. }
14. </style>
16. <!-- Подключение шрифтов -->
17. <link rel="preconnect" href="https://fonts.googleapis.com">
18. <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
19. <link href="https://fonts.googleapis.com/css2?family=Pacifico&family=Yanone+Kaffeesatz:wght@200..700&display=swap" rel="stylesheet">
21. <!-- Внешние стили и скрипты -->
22. <link rel="stylesheet" href="style\_index.css">
23. <script defer src="script.js"></script>
24. </head>
25. <body>
26. <!-- Шапка сайта -->
27. <header>
28. <h1><span id="greeting"></span>, <span id="name"></span></h1>
29. <button class="button-header" onclick="logout()" id="myButton">профиль</button>
30. </header>
31. <!-- Основное содержимое -->
32. <main>
33. <table>
34. <!-- Первая строка: Задачи и Календарь -->
35. <tr>
36. <!-- Блок задач -->
37. <td>
38. <section>
39. <div class="gor" style="margin-bottom: -25px;">
40. <h2>Задачи</h2>
41. <button class="button-add" onclick="add\_task()">+</button>
42. </div>
43. <div class="list"></div>
44. </section>
45. </td>
47. <!-- Блок календаря -->
48. <td>
49. <div class="calendar">
50. <div class="month" id="month"></div>
51. <div class="days" id="days">
52. <div class="day day-name">Пн</div>
53. <div class="day day-name">Вт</div>
54. <div class="day day-name">Ср</div>
55. <div class="day day-name">Чт</div>
56. <div class="day day-name">Пт</div>
57. <div class="day day-name">Сб</div>
58. <div class="day day-name">Вс</div>
59. </div>
60. </div>
61. </td>
62. </tr>
63. <!-- Вторая строка: Привычки и Информация -->
64. <tr>
65. <!-- Блок привычек -->
66. <td>
67. <section>
68. <div class="gor" style="margin-bottom: -25px;">
69. <h2>Привычки</h2>
70. <button class="button-add" onclick="add\_habit()">+</button>
71. </div>
72. <div class="list\_habits"></div>
73. </section>
74. </td>
76. <!-- Блок сегодняшней даты и кнопки -->
77. <td>
78. <table>
79. <tr>
80. <!-- Текущий день -->
81. <td>Сегодня: <span id="day"></span></td>
82. </tr>
83. <tr>
84. <!-- Кнопка информации -->
85. <td colspan="2">
86. <button onclick="go()" style="padding: 10px; font-size: 20px;">О планировании</button>
87. </td>
88. </tr>
89. </table>
90. </td>
91. </tr>
92. </table>
93. </main>
94. <!-- Подвал сайта -->
95. <footer>
96. <p>&copy; 2025 Мой веб-сайт. Все права защищены.</p>
97. </footer>
98. </body>
99. </html>

**enter.html**

1. <!DOCTYPE html>
2. <html lang="ru">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <title>Not to forget</title>
8. <!-- Стили для горизонтального выравнивания -->
9. <style>
10. .gor {
11. display: flex;       /\* расположение элементов по горизонтали \*/
12. align-items: center; /\* выравнивание по вертикали \*/
13. }
14. </style>
16. <!-- Подключение шрифтов -->
17. <link rel="preconnect" href="https://fonts.googleapis.com">
18. <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
19. <link href="https://fonts.googleapis.com/css2?family=Pacifico&family=Yanone+Kaffeesatz:wght@200..700&display=swap" rel="stylesheet">
21. <!-- Внешние стили и скрипты -->
22. <link rel="stylesheet" href="style\_enter.css">
23. <script defer src="script.js"></script>
24. </head>
25. <body>
26. <!-- Основной контейнер форм -->
27. <div class="main">
28. <input type="checkbox" id="chk">
30. <!-- Форма входа -->
31. <div class="login">
32. <form class="form">
33. <label for="chk" aria-hidden="true">Вход</label>
34. <input id="email" class="input" type="email" name="email" placeholder="Email" required>
35. <input id="password" class="input" type="password" name="pswd" placeholder="Password" required>
36. <button type="submit" onclick="Sign\_in(event)">Войти</button>
37. </form>
38. </div>
40. <!-- Форма регистрации -->
41. <div class="register">
42. <form class="form">
43. <label for="chk" aria-hidden="true">Регистрация</label>
44. <input id="username\_reg" class="input" type="text" name="txt" placeholder="Username" required>
45. <input id="email\_reg" class="input" type="email" name="email" placeholder="Email" required>
46. <input id="password\_reg" class="input" type="password" name="pswd" placeholder="Password" required>
47. <button type="submit" onclick="Sign\_up(event)">Зарегистрироваться</button>
48. </form>
49. </div>
50. </div>
51. </body>
52. </html>

**about\_planning.html**

1. <!DOCTYPE html>
2. <html lang="ru">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <title>Планирование привычек и дел</title>
8. <!-- Подключение шрифтов -->
9. <link rel="preconnect" href="https://fonts.googleapis.com">
10. <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
11. <link href="https://fonts.googleapis.com/css2?family=Pacifico&family=Yanone+Kaffeesatz:wght@200..700&display=swap" rel="stylesheet">
13. <!-- Внешние стили и скрипты -->
14. <link rel="stylesheet" href="style\_planning.css">
15. <script defer src="script.js"></script>
16. </head>
17. <body>
18. <!-- Шапка с навигацией -->
19. <header>
20. <nav>
21. <ul>
22. <li><a href="index.html" style="color: white;">Назад</a></li>
23. <li><a href="#habits" style="color: white;">Привычки</a></li>
24. <li><a href="#tasks" style="color: white;">Задачи</a></li>
25. <li><a href="#methods" style="color: white;">Методы</a></li>
26. <li><a href="#resources" style="color: white;">Ресурсы</a></li>
27. </ul>
28. </nav>
29. </header>
30. <!-- Основное содержание -->
31. <main>
32. <!-- Секция формирования привычек -->
33. <section id="habits">
34. <h2>1. Как формировать полезные привычки</h2>
36. <h3>Этапы формирования привычки (по Джеймсу Клиру)</h3>
37. <ol>
38. <li><strong>Сигнал</strong> — триггер, запускающий действие.</li>
39. <li><strong>Желание</strong> — мотивация выполнить действие.</li>
40. <li><strong>Действие</strong> — сама привычка.</li>
41. <li><strong>Награда</strong> — положительное подкрепление.</li>
42. </ol>
43. <h3>Термины</h3>
44. <dl>
45. <dt>Привычка</dt>
46. <dd>Автоматическое действие, сформированное повторением.</dd>
48. <dt>Ритуал</dt>
49. <dd>Последовательность действий, помогающая внедрить привычку.</dd>
50. </dl>
51. </section>
52. <!-- Секция планирования задач -->
53. <section id="tasks">
54. <h2>2. Планирование ежедневных дел</h2>
55. <table>
56. <thead>
57. <tr>
58. <th>Метод</th>
59. <th>Описание</th>
60. <th>Пример</th>
61. </tr>
62. </thead>
63. <tbody>
64. <tr>
65. <td>Матрица Эйзенхауэра</td>
66. <td>
67. <ul>
68. <li><strong>Важно + Срочно</strong> — сделать сразу.</li>
69. <li><strong>Важно + Не срочно</strong> — запланировать.</li>
70. </ul>
71. </td>
72. <td>Делегировать или отложить неважные задачи</td>
73. </tr>
75. <tr>
76. <td>Правило 1-3-5</td>
77. <td>
78. <ul>
79. <li>1 крупная задача</li>
80. <li>3 средние</li>
81. <li>5 мелких</li>
82. </ul>
83. </td>
84. <td>Написать статью (1), ответить на письма (3), убрать стол (5)</td>
85. </tr>
86. </tbody>
87. </table>
88. </section>
89. <!-- Секция методов планирования -->
90. <section id="methods">
91. <h2>3. Популярные методы планирования</h2>
92. <ul>
93. <li><strong>GTD (Getting Things Done)</strong> — система управления задачами.</li>
94. <li><strong>Канбан-доска</strong> — визуализация этапов работы.</li>
95. <li><strong>Помидорро (Pomodoro)</strong> — 25 минут работы / 5 минут отдыха.</li>
96. </ul>
97. </section>
98. <!-- Секция полезных ресурсов -->
99. <section id="resources">
100. <h2>4. Полезные ресурсы</h2>
101. <ul>
102. <li>
103. <a href="https://jamesclear.com/atomic-habits" target="\_blank">
104. Книга "Атомные привычки"
105. </a>
106. </li>
107. </ul>
108. </section>
109. </main>
110. <!-- Подвал сайта -->
111. <footer>
112. <p>&copy; 2025 Мой веб-сайт. Все права защищены.</p>
113. </footer>
114. </body>
115. </html>

**style\_index.css**

1. /\* Стили шапки \*/
2. header {
3. background: #82a8c4;
4. color: #ffffff;          /\* Цвет текста \*/
5. padding: 20px;
6. font-size: 20px;
7. font-family: "Pacifico", cursive;
8. display: flex;           /\* Горизонтальное расположение элементов \*/
9. justify-content: space-between; /\* Распределение пространства между элементами \*/
10. align-items: center;     /\* Вертикальное выравнивание \*/
11. }
12. /\* Основные стили тела документа \*/
13. body {
14. font-family: "Pacifico", cursive;
15. }
16. /\* Заголовки второго уровня \*/
17. h2 {
18. font-size: 30px;
19. }
20. /\* Стилизация таблицы \*/
21. table {
22. width: 100%;            /\* Полная ширина контейнера \*/
23. border-collapse: collapse; /\* Объединение границ ячеек \*/
24. }
25. /\* Стили ячеек таблицы \*/
26. th, td {
27. padding: 10px;          /\* Внутренние отступы \*/
28. text-align: left;       /\* Выравнивание текста по левому краю \*/
29. }
30. /\* Стили состояний дней привычек \*/
31. .h4-now {
32. background-color: #fff;
33. color: #2c3e50;
34. border-radius: 10px;
35. padding: 5%;            /\* Текущий день \*/
36. }
37. .h4-future {
38. background-color: #fff;
39. color: #2c3e50;
40. border-radius: 10px;
41. padding: 5%;            /\* Будущие дни \*/
42. }
43. .h4-red {
44. background-color: #fff;
45. color: #e67e22;
46. border-radius: 10px;
47. padding: 5%;            /\* Просроченные дни \*/
48. }
49. .h4-green {
50. background-color: #fff;
51. color: #27ae60;
52. border-radius: 10px;
53. padding: 5%;            /\* Выполненные дни \*/
54. }
55. /\* Базовые стили кнопок \*/
56. button {
57. font-family: "Yanone Kaffeesatz", sans-serif;
58. background-color: #2c3e50;
59. color: #ffffff;
60. border-radius: 5px;
61. cursor: pointer;        /\* Указатель при наведении \*/
62. }
63. /\* Стили кнопки в шапке \*/
64. .button-header {
65. width: 130px;
66. height: 40px;
67. padding: 10px 25px;
68. font-size: 25px;
69. font-weight: 400;
70. position: relative;
71. border: none;
72. background-color: #2c3e50;
73. color: #ffffff;
74. overflow: hidden;       /\* Для анимации \*/
75. }
76. /\* Анимация "свечения" кнопки \*/
77. .button-header:before {
78. position: absolute;
79. content: '';
80. top: -180px;
81. left: 0;
82. width: 30px;
83. height: 100%;
84. background-color: #fff;
85. animation: anim 5s ease-in-out infinite; /\* Бесконечная анимация \*/
86. }
87. /\* Состояния кнопки \*/
88. .button-header:hover {
89. opacity: .7;            /\* Эффект при наведении \*/
90. }
91. .button-header:active {
92. box-shadow: 4px 4px 6px 0 rgba(255,255,255,.3),
93. -4px -4px 6px 0 rgba(116, 125, 136, .2),
94. inset -4px -4px 6px 0 rgba(255,255,255,.2),
95. inset 4px 4px 6px 0 rgba(0, 0, 0, .2); /\* Эффект нажатия \*/
96. }
97. /\* Ключевые кадры анимации \*/
98. @keyframes anim {
99. 0% { -webkit-transform: scale(0) rotate(45deg); opacity: 0; }
100. 80% { -webkit-transform: scale(0) rotate(45deg); opacity: 0.5; }
101. 81% { -webkit-transform: scale(4) rotate(45deg); opacity: 1; }
102. 100% { -webkit-transform: scale(50) rotate(45deg); opacity: 0; }
103. }
104. /\* Стили подвала \*/
105. footer {
106. text-align: center;
107. padding: 10px;
108. background-color: #82a8c4;
109. }
110. /\* Кнопка добавления \*/
111. .button-add {
112. padding: 1px 5px;
113. margin-left: 10px;
114. border: none;
115. border-radius: 30%;     /\* Круглая форма \*/
116. background-color: #2c3e50;
117. color: #ffffff;
118. }
119. /\* Блоки задач/привычек \*/
120. .block {
121. background-color: #7f8c8d;
122. margin: 30px;
123. font-family: "Yanone Kaffeesatz", sans-serif;
124. color: #fff;
125. font-size: 20px;
126. padding: 5px 40px;
127. border-radius: 10px;    /\* Скругленные углы \*/
128. }
129. /\* Стили календаря \*/
130. .calendar {
131. border: 1px solid #ccc;
132. border-radius: 8px;
133. overflow: hidden;       /\* Скрытие выходящего контента \*/
134. }
135. .month {
136. background-color: #f2f2f2;
137. text-align: center;
138. padding: 1px;
139. font-size: 24px;        /\* Заголовок месяца \*/
140. }
141. .days {
142. display: grid;          /\* Сетка для дней \*/
143. grid-template-columns: repeat(7, 1fr); /\* 7 колонок \*/
144. border-top: 1px solid #ccc;
145. }
146. .day {
147. border: 1px solid #ccc;
148. padding: 2px;
149. text-align: center;     /\* Ячейки дней \*/
150. }
151. .day-name {
152. background-color: #e0e0e0;
153. font-weight: bold;      /\* Заголовки дней недели \*/
154. }

**style\_enter.css**

1. /\* Основной контейнер форм \*/
2. .main {
3. position: relative;
4. display: flex;
5. justify-self: center;     /\* Выравнивание по оси X (возможна опечатка) \*/
6. flex-direction: column;   /\* Расположение элементов в колонку \*/
7. background-color: #82a8c4;
8. max-height: 540px;        /\* Максимальная высота контейнера \*/
9. overflow: hidden;         /\* Скрытие выходящего за пределы контента \*/
10. border-radius: 12px;      /\* Скругление углов \*/
11. box-shadow: 7px 7px 10px 3px #2c3e50; /\* Тень контейнера \*/
12. }
13. /\* Стили для всех кнопок \*/
14. button {
15. font-family: "Yanone Kaffeesatz", sans-serif;
16. background-color: #2c3e50;
17. color: #ffffff;
18. border-radius: 5px;
19. cursor: pointer;          /\* Курсор-указатель при наведении \*/
20. }
21. /\* Основные стили страницы \*/
22. body {
23. font-family: "Pacifico", cursive;
24. display: flex;
25. justify-content: center;  /\* Центрирование по горизонтали \*/
26. align-items: center;      /\* Центрирование по вертикали \*/
27. height: 100vh;           /\* Полноэкранная высота \*/
28. margin: 0;               /\* Удаление стандартных отступов \*/
29. }
30. /\* Стилизация формы \*/
31. .form {
32. display: flex;
33. flex-direction: column;  /\* Элементы формы в колонку \*/
34. gap: 14px;               /\* Расстояние между элементами \*/
35. padding-left: 44px;
36. padding-right: 44px;
37. padding-bottom: 64px;
38. }
39. /\* Скрытие чекбокса переключения форм \*/
40. #chk {
41. display: none;
42. }
43. /\* Стили для формы входа \*/
44. .login label {
45. margin: 25% 0 5%;       /\* Внешние отступы для заголовка \*/
46. }
47. /\* Общие стили для заголовков форм \*/
48. label {
49. color: #fff;
50. font-size: 250%;         /\* Крупный размер текста \*/
51. justify-content: center;
52. display: flex;
53. font-weight: bold;
54. cursor: pointer;         /\* Курсор-указатель \*/
55. transition: .5s ease-in-out; /\* Плавная анимация \*/
56. }
57. /\* Стилизация полей ввода \*/
58. .input {
59. height: 40px;
60. background: #e0dede;     /\* Цвет фона \*/
61. padding: 10px;
62. border: none;
63. outline: none;           /\* Удаление контура при фокусе \*/
64. border-radius: 4px;
65. }
66. /\* Стили для формы регистрации \*/
67. .register {
68. display: flex;
69. justify-content: center;
70. align-items: center;
71. background: #eee;        /\* Цвет фона \*/
72. border-radius: 60% / 10%; /\* Нестандартное скругление \*/
73. transform: translateY(5%); /\* Начальное положение \*/
74. transition: .8s ease-in-out; /\* Анимация переключения \*/
75. }
76. .register label {
77. color: #2c3e50;          /\* Цвет текста \*/
78. transform: scale(.6);    /\* Начальный масштаб \*/
79. }
80. /\* Анимация при активации переключателя \*/
81. #chk:checked ~ .register {
82. transform: translateY(-60%); /\* Сдвиг формы регистрации \*/
83. }
84. #chk:checked ~ .register label {
85. transform: scale(1);     /\* Нормальный масштаб текста \*/
86. margin: 10% 0 5%;        /\* Внешние отступы \*/
87. }
88. #chk:checked ~ .login label {
89. transform: scale(.6);    /\* Уменьшение текста входа \*/
90. margin: 5% 0 5%;         /\* Корректировка отступов \*/
91. }
92. /\* Стилизация кнопок формы \*/
93. .form button {
94. width: 85%;              /\* Ширина кнопки \*/
95. height: 40px;
96. margin: 12px auto 10%;   /\* Центрирование кнопки \*/
97. color: #fff;
98. background: #2c3e50;
99. font-size: 1rem;
100. font-weight: bold;
101. border: none;
102. border-radius: 4px;
103. cursor: pointer;
104. transition: .2s ease-in; /\* Плавное изменение цвета \*/
105. }
106. /\* Эффект при наведении на кнопку \*/
107. .form button:hover {
108. background-color: #2c3e50; /\* Постоянный цвет (возможно требуется изменение) \*/
109. }

**style\_planning.css**

1. /\* Стили шапки \*/
2. header {
3. padding: 20px;                 /\* Внутренние отступы \*/
4. font-size: 25px;               /\* Размер шрифта \*/
5. font-family: "Pacifico", cursive; /\* Шрифт заголовка \*/
6. display: flex;                 /\* Горизонтальное расположение элементов \*/
7. justify-content: space-around; /\* Равномерное распределение пространства \*/
8. }
9. /\* Основные стили страницы \*/
10. body {
11. font-size: 25px;               /\* Базовый размер текста \*/
12. background-color: #82a8c4;     /\* Цвет фона страницы \*/
13. font-family: "Yanone Kaffeesatz", sans-serif; /\* Основной шрифт \*/
14. }
15. /\* Навигационное меню \*/
16. nav ul {
17. list-style: none;             /\* Убираем маркеры списка \*/
18. display: flex;                /\* Горизонтальное расположение пунктов \*/
19. justify-content: center;      /\* Центрирование элементов \*/
20. gap: 20px;                   /\* Расстояние между пунктами меню \*/
21. }
22. /\* Стили секций контента \*/
23. section {
24. background: #fff;             /\* Белый фон секций \*/
25. margin: 20px 0;               /\* Внешние отступы сверху/снизу \*/
26. padding: 20px;                /\* Внутренние отступы \*/
27. border-radius: 5px;           /\* Скругление углов \*/
28. }
29. /\* Общие стили кнопок \*/
30. button {
31. font-family: "Yanone Kaffeesatz", sans-serif; /\* Шрифт кнопок \*/
32. background-color: #2c3e50;    /\* Цвет фона \*/
33. color: #ffffff;               /\* Цвет текста \*/
34. padding: 10px;                /\* Внутренние отступы \*/
35. font-size: 20px;              /\* Размер текста \*/
36. border-radius: 5px;           /\* Скругление углов \*/
37. cursor: pointer;              /\* Изменение курсора при наведении \*/
38. }
39. /\* Стилизация таблиц \*/
40. table {
41. width: 100%;                  /\* Полная ширина контейнера \*/
42. border-collapse: collapse;    /\* Объединение границ ячеек \*/
43. }
44. /\* Стили ячеек таблицы \*/
45. th, td {
46. padding: 12px;                /\* Внутренние отступы \*/
47. border: 1px solid #ddd;       /\* Границы ячеек \*/
48. }
49. /\* Стили заголовков таблицы \*/
50. th {
51. background: #82a8c4;          /\* Цвет фона заголовков \*/
52. color: white;                 /\* Цвет текста \*/
53. font-size: 20px;              /\* Размер шрифта \*/
54. }
55. /\* Стили подвала \*/
56. footer {
57. text-align: center;           /\* Центрирование текста \*/
58. padding: 10px;                /\* Внутренние отступы \*/
59. }

**script.js**

1. // Глобальные переменные состояния
2. let name = "гость";
3. let sign = false;
4. let registered = {}; // Хранилище зарегистрированных пользователей
5. // Элементы интерфейса
6. const username = document.getElementById("name");
7. username.textContent = name;
8. const div = document.querySelector(".list");
9. const div\_habits = document.querySelector(".list\_habits");
10. // Функция выхода из системы
11. function logout() {
12. window.location.href = "enter.html";
13. }
14. // Обработчик входа
15. function Sign\_in(event) {
16. event.preventDefault();
18. const login = document.getElementById("email");
19. const password = document.getElementById("password");
21. if (login.value in registered && password.value === registered[login.value][0]) {
22. sign = true;
23. window.location.href = "index.html";
24. } else {
25. alert("Неверный логин или пароль");
26. }
28. login.value = "";
29. password.value = "";
30. }
31. // Обработчик регистрации
32. function Sign\_up(event) {
33. event.preventDefault();
34. const username = document.getElementById("username\_reg");
35. const login = document.getElementById("email\_reg");
36. const password = document.getElementById("password\_reg");
37. if (login.value in registered) {
38. alert("Пользователь уже зарегистрирован");
39. } else {
40. registered[login.value] = [password.value, username.value];
41. alert("Успешная регистрация");
42. }
44. username.value = "";
45. login.value = "";
46. password.value = "";
47. }
48. // Переход на страницу о планировании
49. function go() {
50. window.location.href = "about\_planning.html";
51. }
52. // Удаление элемента
53. const Delete = (event) => {
54. event.target.parentNode.remove();
55. }
56. // Отметка выполнения привычки
57. const Notice = (event) => {
58. const habitBlock = event.target.closest('.block');
59. const daysContainer = habitBlock.querySelectorAll('.gor')[1];
60. const todayElement = daysContainer.querySelector('.h4-now');
61. todayElement.classList.replace('h4-now', 'h4-green');
62. }
63. // Добавление новой задачи
64. function add\_task() {
65. const task = prompt("Введите задачу:");
66. if (!task) return;
68. const date = prompt("Введите дату (DD-MM-YYYY):");
69. if (!date) return;
70. if (!/^\d{2}-\d{2}-\d{4}$/.test(date)) {
71. alert("Некорректная дата! Используйте формат DD-MM-YYYY");
72. return;
73. }
74. const newTask = document.createElement("div");
75. newTask.innerHTML += `
76. <div class="block">
77. <h3>${task}</h3>
78. <p>${date}</p>
79. <button onclick="Delete(event)">удалить</button>
80. </div>`;
81. div.append(newTask);
82. }
83. // Работа с датами
84. const date = new Date();
85. const hours = date.getHours();
86. const dayIndex = date.getDay();
87. const daysOfWeek = ["Воскресенье", "Понедельник", "Вторник", "Среда", "Четверг", "Пятница", "Суббота"];
88. const dayName = daysOfWeek[dayIndex]
89. document.getElementById("day").textContent = dayName;
90. // Добавление новой привычки
91. function add\_habit() {
92. const task = prompt("Введите привычку:");
93. if (!task) return;
94. const newTask = document.createElement("div");
95. let flag = false;
96. const a = [];
98. // Формирование классов для дней недели
99. const reorderedDays = [...daysOfWeek.slice(1), ...daysOfWeek.slice(0, 1)];
100. for (const day of reorderedDays) {
101. if (!flag && dayName !== day) {
102. a.push("h4-red");
103. } else if (dayName === day) {
104. flag = true;
105. a.push("h4-now");
106. } else {
107. a.push("h4-future");
108. }
109. }
110. newTask.innerHTML += `
111. <div class="block">
112. <div class="gor">
113. <h3 style="margin-bottom: -15px;">${task}</h3>
114. <button style="background-color: #fff; color: black; margin-left: 4%;"
115. onclick="Notice(event)">
116. отметить выполнение
117. </button>
118. </div>
119. <div class="gor" style="justify-content: space-between;">
120. ${a.map((cls, i) => `<h4 class="${cls}">${['пн','вт','ср','чт','пт','сб','вс'][i]}</h4>`).join('')}
121. </div>
122. <button onclick="Delete(event)">удалить</button>
123. </div>`;
124. div\_habits.append(newTask);
125. }
126. // Определение приветствия
127. const greetings = {
128. morning: "Доброе утро",
129. night: "Доброй ночи",
130. afternoon: "Добрый день",
131. evening: "Добрый вечер"
132. };
133. let greetingText;
134. if (hours < 6) {
135. greetingText = greetings.night;
136. } else if (hours < 12) {
137. greetingText = greetings.morning;
138. } else if (hours < 16) {
139. greetingText = greetings.afternoon;
140. } else if (hours < 22) {
141. greetingText = greetings.evening;
142. } else {
143. greetingText = greetings.night;
144. }
145. document.getElementById("greeting").textContent = greetingText;
146. // Генерация календаря
147. const now = new Date();
148. const month = now.getMonth();
149. const year = now.getFullYear();
150. const monthNames = [
151. "Январь", "Февраль", "Март", "Апрель", "Май", "Июнь",
152. "Июль", "Август", "Сентябрь", "Октябрь", "Ноябрь", "Декабрь"
153. ];
154. document.getElementById('month').innerText = `${monthNames[month]} ${year}`;
155. const firstDay = new Date(year, month, 1).getDay();
156. const daysInMonth = new Date(year, month + 1, 0).getDate();
157. const daysContainer = document.getElementById('days');
158. // Очистка предыдущего календаря
159. daysContainer.innerHTML = '';
160. // Заполнение пустых ячеек
161. for (let i = 0; i < (firstDay === 0 ? 6 : firstDay - 1); i++) {
162. daysContainer.innerHTML += '<div class="day"></div>';
163. }
164. // Заполнение днями месяца
165. for (let day = 1; day <= daysInMonth; day++) {
166. const dayCell = document.createElement('div');
167. dayCell.className = 'day';
168. dayCell.innerText = day;
169. daysContainer.appendChild(dayCell);
170. }