**5 Подключение библиотеки jQuery к приложению и получение доступа к элементам документа. Работа с элементами DOM при помощи библиотеки jQuery. Создание анимации при помощи библиотеки jQuery**

**Цель:** Ознакомление с подключеним библиотеки jQuery к приложению и полученим доступа к элементам документа. Изучить официальную документацию по библиотеке. Изучить организацию работы с элементами DOM при помощи библиотеки jQuery. Ознакомиться с особенностями работы библиотеки AOS.js. Реализовать анимации на странице сайта с использованием библиотеки AOS.js.

На начальном этапе практики я изучил основы подключения и настройки библиотеки AOS.js, предназначенной для добавления анимаций на веб-страницы. Я подключил необходимые CSS и JavaScript файлы, а также создал базовый HTML-шаблон для использования этих анимаций. Далее, я реализовал инициализацию простейших анимаций с помощью базовых настроек конфигурации, а затем перешел к более продвинутым возможностям библиотеки. Я настроил различные виды анимаций, такие как fade, flip, zoom, slide и другие. Также я управлял направлением, продолжительностью и задержкой анимаций, чтобы достичь желаемого эффекта.. Пример объектов на которые была применена анимация представлены на рисунке 7.1:

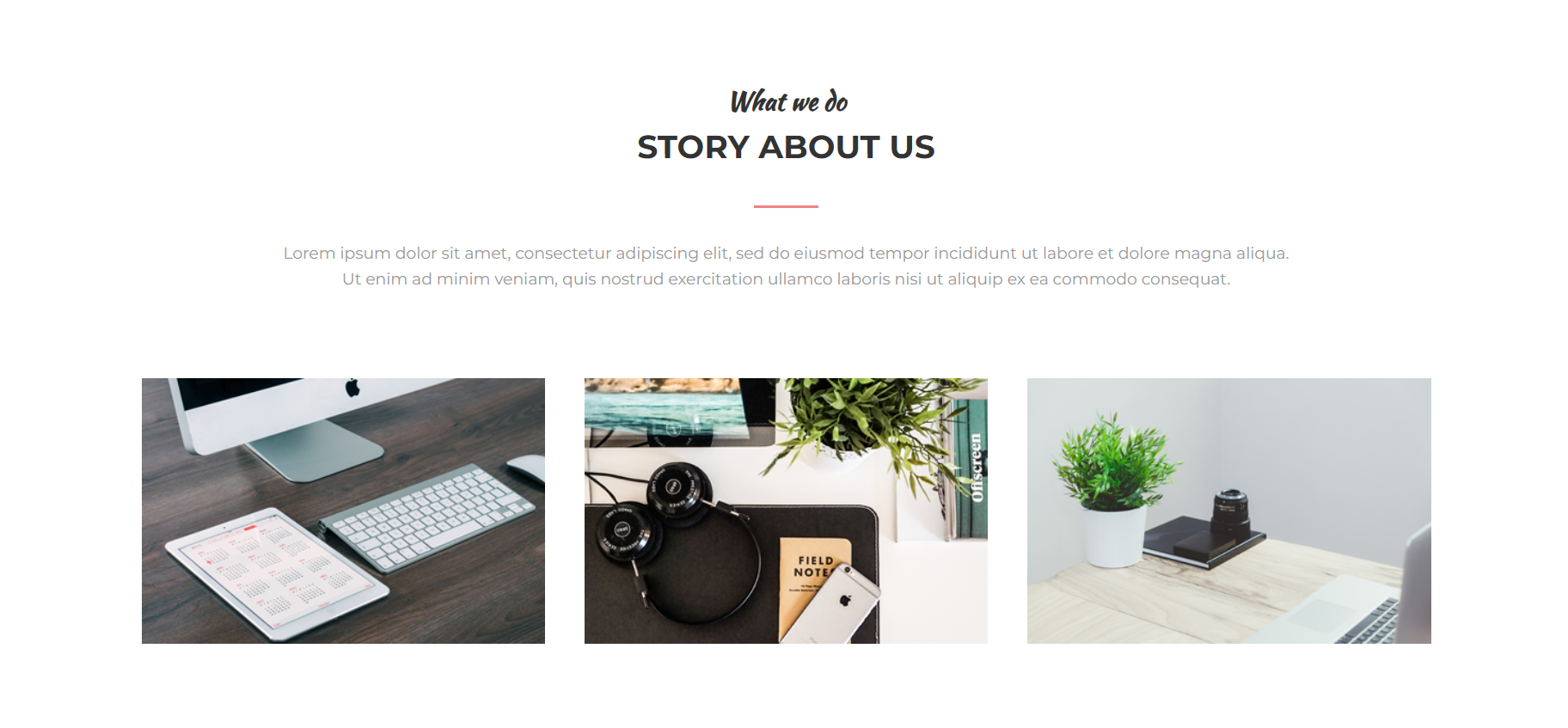


Рисунок 5.1 – Элементы с анимацией

Источник: собственная разработка

Листинг кода:

<section class="section" id="about">

        <div class="container">

            <div class="section\_\_header">

                <h3 class="section\_\_suptitle">What we do</h3>

                <h2 class="section\_\_title">Story about us</h2>

                <div class="section\_\_text">

                    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco nisi ut aliquip ex ea commodo consequat.</p>

                </div>

            </div>

            <div class="card"  data-aos="fade-up" data-aos-duration="2000" data-aos-delay="100">

                <div class="card\_\_item">

                    <div class="card\_\_inner">

                        <div class="card\_\_img">

                            <img src="assets/about\_us/worktableleft.png" alt="Worktable">

                        </div>

                        <div class="card\_\_text">

                            <img src="assets/about\_us/users.png" alt="users">

                            <br>

                            super team

                        </div>

                    </div>

                </div>

                <div class="card\_\_item">

                    <div class="card\_\_inner">

                        <div class="card\_\_img">

                            <img src="assets/about\_us/worktablecenter.png" alt="Worktable">

                        </div>

                        <div class="card\_\_text">

                            <img src="assets/about\_us/users.png" alt="users">

                            <br>

                            super team

                        </div>

                    </div>

                </div>

                <div class="card\_\_item">

                    <div class="card\_\_inner">

                        <div class="card\_\_img">

                            <img src="assets/about\_us/worktableright.png" alt="Worktable">

                        </div>

                        <div class="card\_\_text">

                            <img src="assets/about\_us/users.png" alt="users">

                            <br>

                            super team

                        </div>

                    </div>

                </div>

            </div>

        </div>

    </section>

**Контрольные вопросы:**

1. Для чего используются сторонние библиотеки?

Сторонние библиотеки используются для ускорения разработки, добавления функциональности без написания кода с нуля и решения распространенных задач (например, манипуляция DOM, AJAX-запросы, анимации).

2. Укажите назначение следующих методов (команд) библиотеки jQuery: html(), val(), find(), append(), attr(). Приведите аналог этих команд на синтаксисе современного JavaScript.

Назначение методов jQuery и их аналоги в современном JavaScript:

html(): Получение или установка HTML-содержимого.

// jQuery

$('element').html('new content');

// JS

document.querySelector('element').innerHTML = 'new content';

val(): Получение или установка значения формы.

// jQuery

$('input').val('new value');

// JS

document.querySelector('input').value = 'new value';

find(): Поиск потомков элемента.

// jQuery

$('element').find('child');

// JS

document.querySelector('element').querySelectorAll('child');

append(): Добавление содержимого в конец элемента.

// jQuery

$('element').append('<div>new element</div>');

// JS

document.querySelector('element').insertAdjacentHTML('beforeend', '<div>new element</div>');

attr(): Получение или установка атрибута.

// jQuery

$('element').attr('attribute', 'value');

// JS

document.querySelector('element').setAttribute('attribute', 'value');

3. Перечислите и опишите работу методов jQuery, предназначенных для работы с css классами.

Методы jQuery для работы с CSS-классами:

addClass(): добавляет один или несколько классов;

removeClass(): удаляет один или несколько классов;

toggleClass(): переключает класс (если есть, удаляет; если нет, добавляет);

hasClass(): проверяет наличие класса у элемента.

4. Укажите назначение библиотеки AOS.js.

AOS (Animate On Scroll) — библиотека для анимации элементов при прокрутке страницы.

5. Назовите способы подключения библиотеки AOS.js к странице.

Способы подключения библиотеки AOS.js к странице:

через CDN:

<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/aos@2.3.4/dist/aos.css" />

<script src="https://cdn.jsdelivr.net/npm/aos@2.3.4/dist/aos.js"></script>

локально, скачав файлы и подключив их:

<link rel="stylesheet" href="path/to/aos.css" />

<script src="path/to/aos.js"></script>

6. Опишите процесс использования библиотеки AOS.js

Процесс использования библиотеки AOS.js:

подключить библиотеку к проекту;

инициализировать AOS при помощи AOS.init().

7. Назовите основные настройки, применяемые для управления видом анимации элементов с помощью AOS.js

Основные настройки AOS.js для управления видом анимации:

data-aos: тип анимации (например, `fade-up`, `fade-down`);

data-aos-duration: длительность анимации (в миллисекундах);

data-aos-delay: задержка перед началом анимации (в миллисекундах);

data-aos-offset: смещение (в пикселях) для начала анимации;

data-aos-easing: тип ускорения (например, `ease`, `ease-in-out`).