

Учреждение образования  
«Белорусский государственный технологический университет»

Факультет информационных технологий  
Кафедра информационных систем и технологий  
Специальность 6-05-0611-01 Информационные системы и технологии

ОТЧЕТ  
по учебной ознакомительной практике

Исполнитель

студент 1 курса 1 группы

\_\_\_\_\_  
(подпись, дата)

Жарский.А.В

\_\_\_\_\_  
(Ф.И.О.)

Руководитель практики  
от кафедры

старший преподаватель

\_\_\_\_\_  
(должность, уч. звание)

\_\_\_\_\_  
(подпись, дата)

Барковский Е.В.

\_\_\_\_\_  
(Ф.И.О.)

Отчет защищен с оценкой \_\_\_\_\_

Минск 2024 г.

## Оглавление

1. Постановка задачи .....	3
1.1 Техническое задание .....	3
1.2 Выбор средств реализации программного продукта .....	3
2. Реализация веб-сайта.....	5
2.1 Оформление веб-сайта.....	5
2.1.1 Выбор способа вёрстки.....	5
2.1.2 Выбор шрифтового оформления .....	6
2.2 Реализация основных элементов веб-сайта.....	6
2.2.1 Реализация «шапки» сайта .....	6
2.2.2 Реализация «тела» сайта .....	6
3. Реализация структуры программы.....	9

## **1. Постановка задачи**

### **1.1 Техническое задание**

В рамках учебной практики была поставлена задача разработать многостраничный веб-сайт, который будет предоставлять пользователям возможность генерировать случайные цвета в формате RGB и получать информацию о них. Сайт должен быть простым в использовании и иметь интуитивно понятный интерфейс. Веб-сайт должен помочь пользователям легко находить и использовать различные оттенки для своих проектов.

Основными задачами являются:

- разработка пользовательского интерфейса.
- реализация функционала генерации случайных цветов.
- Основной задачей является создание привлекательного и стильного интерфейса сайта, с удачным размещением всех элементов на страницах. Все компоненты будут грамотно организованы, чтобы обеспечить понятность и удобство использования для целевой аудитории. Текущего сгенерированный цвет должен отображаться в виде блока соответствующего цвета и его кода в формате RGB.

Кроме того, веб-сайт будет разработан с учётом кроссбраузерности, что означает, что он будет корректно отображаться в различных браузерах. Также сайт будет адаптивным, что позволит пользователям просматривать его на мобильных устройствах, планшетах и ноутбуках. Это обеспечит возможность доступа к сайту с любого устройства, что является важным фактором для привлечения и удержания пользователей.

### **1.2 Выбор средств реализации программного продукта**

Поставленная задача выполняется в редакторе кода Visual Studio Code, который представляет собой упрощенный, но мощный редактор исходного кода, работающий на компьютере и доступный для операционных систем Windows, macOS и Linux. Visual Studio Code обладает встроенной поддержкой JavaScript, TypeScript и Node.js, а также предлагает обширную экосистему расширений для других языков программирования и сред выполнения. Это делает его идеальным выбором для разработки веб-приложений различной сложности. При создании сайта и его функционала будут использованы React js, CSS и HTML.

React – это JavaScript-библиотека для разработки пользовательских интерфейсов, которая позволяет создавать компоненты, обеспечивая модульную и масштабируемую разработку веб-приложений. React использует виртуальный DOM и эффективный механизм обновления компонентов, что способствует созданию быстрых и отзывчивых пользовательских интерфейсов.

JavaScript (JS) – это язык программирования, который широко используется для создания интерактивных и динамических веб-страниц. Он является одним из основных языков разработки веб-приложений и поддерживается всеми современными веб-браузерами. JavaScript позволяет добавлять сложные

функциональные возможности к веб-сайтам, делая их более динамичными и взаимодействующими с пользователем.

CSS (Cascading Style Sheets) – это язык стилей, используемый для описания внешнего вида элементов веб-страницы. С помощью CSS можно определить цвета, шрифты, размеры, расположение и другие аспекты стилизации элементов. CSS играет ключевую роль в создании привлекательного и согласованного внешнего вида веб-приложений, позволяя разработчикам контролировать визуальное представление контента.

HTML (Hypertext Markup Language) – это язык разметки, используемый для создания структуры и содержимого веб-страницы. Он определяет различные элементы, такие как заголовки, параграфы, изображения, таблицы и другие, которые составляют содержимое веб-страницы. HTML является фундаментом любой веб-страницы и используется для создания структуры и содержимого веб-приложений.

Сочетание данных технологий и инструментов позволило создать интерактивное веб-приложение с привлекательным дизайном и плавной навигацией. Благодаря использованию Visual Studio Code, разработчики могут эффективно работать над проектом, используя мощные инструменты для написания кода, отладки и тестирования. React js обеспечивает высокую производительность и масштабируемость приложения, в то время как CSS и HTML гарантируют его визуальную привлекательность и структурированность. В результате получается современное веб-приложение, способное удовлетворить потребности пользователей и предоставить им удобный интерфейс для взаимодействия с сайтом.

## **2. Реализация веб-сайта**

### **2.1 Оформление веб-сайта**

#### **2.1.1 Выбор способа вёрстки**

В поставленной задаче разработки веб-сайта было решено использовать flex-вёрстку для реализации кроссбраузерности и адаптивности. Flex-вёрстка предоставляет гибкость в организации и выравнивании элементов на странице, что позволяет создавать интерфейсы, оптимизированные для различных размеров экранов и устройств. Это особенно важно в эпоху растущего числа мобильных устройств и различных размеров экранов, когда одинаково хорошо выглядеть и функционировать должны как на больших мониторах, так и на маленьких смартфонах.

Кроме того, flex-вёрстка упрощает управление направлением элементов (по строкам или столбцам), их выравниванием по горизонтали и вертикали, а также распределением свободного пространства между ними. Таким образом, выбор flex-вёрстки для разработки веб-сайта обеспечивает не только его кроссбраузерность и адаптивность, но и упрощает процесс разработки, делая код более читаемым и легким для поддержки.

#### **2.1.2 Выбор стилового оформления**

Для данного сайта выбран стиль минимализма, который характеризуется упрощением дизайна, фокусом на ключевых элементах и созданием чистого и элегантного внешнего вида. Этот подход позволяет сосредоточиться на самом важном, удаляя все лишнее, что может отвлекать внимание пользователя от основного содержания.

Минималистский дизайн способствует лучшей восприимчивости информации благодаря отсутствию излишеств и перегруженности элементами. Преимущества такого стиля включают ясность и простоту структуры, что облегчает навигацию по сайту и улучшает пользовательский опыт. Минимализм также способствует привлекательному внешнему виду, делая сайт более современным и профессиональным.

Одним из значительных преимуществ минималистского дизайна является быстрая загрузка страницы, что достигается за счет уменьшения количества ресурсов, необходимых для отображения страницы. Это особенно важно в условиях медленных интернет-соединений или при использовании мобильных устройств, где скорость загрузки может существенно повлиять на удовлетворенность пользователя. Выбор минимализма для сайта генерации цветов в формате RGB подчеркивает функциональность и удобство использования сервиса.

Такой подход позволяет пользователям быстро найти нужную информацию и воспользоваться предложенными функциями без необходимости тратить время на изучение сложного интерфейса. Кроме того, минималистский дизайн помогает акцентировать внимание на самих цветах и их генерации, делая процесс более интуитивным и приятным.

### 2.1.2 Выбор шрифтового оформления

Для оформления текстового содержания сайта был выбран шрифт Ubuntu Light. Шрифт Ubuntu Light зарекомендовал себя как популярный и универсальный вариант для оформления текстового содержания на веб-сайтах. Его особенности включают в себя современный стиль, отличную читаемость и широкую применимость, что делает его идеальным выбором для создания визуально привлекательных и функциональных веб-проектов.

Ubuntu Light отличается своей легкостью и свежестью, что делает текст на сайте более доступным и приятным для глаз. Этот шрифт идеально подходит для цифровых платформ, где важна читаемость и удобство восприятия информации. Его современные черты гармонично сочетаются с классическими элементами, делая шрифт подходящим для широкого спектра проектов, от корпоративных сайтов до творческих портфолио и образовательных платформ.

Одним из ключевых преимуществ шрифта Ubuntu Light является его высокая читаемость даже при небольших размерах, что особенно важно для мобильных устройств и людей с ограниченными возможностями зрения. Это обеспечивает комфорт чтения для большого числа пользователей, улучшая общее впечатление от посещения сайта.

## 2.2 Реализация основных элементов веб-сайта

### 2.2.1 Реализация «шапки» сайта

Реализация хедера веб-сайта включает создание верхней части страницы.



Рисунок 2.1 – «Шапка» веб-сайта

На рисунке 2.1 показана «шапка», которая содержит название сайта и кнопку генерации цвета.

### 2.2.2 Реализация «тела» сайта

Тело веб-страницы состоит из нескольких частей: инструмент для генерации цвета, кнопок, позволяющих выбрать заранее установленные цвета, дополнительная информация, предоставляющая пользователю сведения о том, как работает RGB.



Рисунок 2.2 – Панель инструментов

Панель инструментов содержит три слайдера, отвечающих за три цвета: красный, зелёный и синий. Снизу находится блок, отображающий выбранный цвет. Каждый из этих слайдеров позволяет пользователю тонко настраивать оттенок выбранного цвета, обеспечивая высокую точность и гибкость при генерации цветовых комбинаций.

Блок внизу, кроме отображения выбранного цвета, также предоставляет пользователю возможность скопировать код цвета в буфер обмена с помощью одного клика. Это значительно упрощает процесс применения сгенерированных цветов в других проектах или приложениях.

Также стоит отметить, что интерфейс панели инструментов разработан таким образом, чтобы быть максимально интуитивно понятным даже для новых пользователей. Все элементы управления размещены логично и доступны для быстрого доступа, что делает процесс генерации и выбора цветов достаточно простым.

## RGB — это аббревиатура, которая означает Red Green Blue (Красный Зеленый Синий)

Это модель цветопередачи, используемая в электронных устройствах для отображения цветов на экранах. В RGB модели каждый цвет представляется комбинацией красного, зеленого и синего цветов в различных пропорциях. Эта модель широко используется в цифровых камерах, компьютерах, телевизорах и других устройствах с цветным дисплеем.

### История RGB

История использования RGB модели начинается с изобретения первых электронных устройств для отображения изображений. В начале XX века были разработаны первые экспериментальные системы цветного телевидения, которые использовали принципы, лежащие в основе современной RGB модели. Однако широкое распространение эта модель получила только с развитием компьютерной графики и появлением персональных компьютеров в конце XX века.

### Как работает RGB модель?

В RGB модели каждый пиксель на экране представляет собой комбинацию красного, зеленого и синего света. Интенсивность каждого из этих цветов может варьироваться от 0 до 255, что позволяет создать более 16 миллионов различных оттенков цвета. Комбинация этих трех основных цветов позволяет воспроизвести практически любой видимый человеком цвет.

**Красный (Red):** Определяет интенсивность красного цвета в пикселе. Значение может варьироваться от 0 (нет красного) до 255 (максимальная интенсивность красного).

**Зеленый (Green):** Аналогично определяет интенсивность зеленого цвета. Значения также варьируются от 0 до 255.

**Синий (Blue):** Определяет интенсивность синего цвета. Значения варьируются от 0 до 255.

### Примеры использования RGB

RGB модель используется во многих областях, включая:

**Веб-дизайн:** Для определения цветов элементов на веб-страницах разработчики используют RGB значения.

**Графический дизайн:** Программы для редактирования изображений, такие как Adobe Photoshop, используют RGB модель для работы с цветом.

**Цифровая фотография:** Камеры и программное обеспечение для обработки фотографий часто работают с RGB изображениями.

**Производство телевизоров и мониторов:** Производители используют RGB модель для настройки и контроля качества цветопередачи.

### Заключение

RGB модель является основой современной цветовой передачи в электронных устройствах. Благодаря ей мы можем воспринимать миллионы оттенков цветов на экранах наших устройств, что делает возможным создание реалистичных изображений и передачу точного цвета в различных приложениях.

## Рисунок 2.3 – Раздел с информацией о RGB

Такое описание поможет неосведомлённым пользователям быстрее разобраться в том, как работает веб-сайт, а затем применять его функции в своих целях.



### 3. Реализация структуры программы

Реализация основывалась на использовании хуков состояния (useState) для управления значениями красного, зеленого и синего компонентов цвета. Компонент ColorSlider представляет собой функциональный компонент, который включает в себя три слайдера для выбора значений RGB, кнопку для генерации случайного цвета и набор предустановленных цветовых схем.

```
import React, { useState } from 'react';
import Content from './Content'
import Slider from 'react-input-slider';

const ColorSlider = () => {
  const [red, setRed] = useState(0);
  const [green, setGreen] = useState(0);
  const [blue, setBlue] = useState(0);

  const rgbColor = `rgb(${red}, ${green}, ${blue})`;

  const changeColor = (r,g,b) => {
    setRed(r);
    setGreen(g);
    setBlue(b);
  }

  const copyToClipboard = () => {
    navigator.clipboard.writeText(rgbColor).then(() => {
      alert('Цвет скопирован в буфер обмена');
    }, (err) => {
      console.error('Ошибка копирования цвета:', err);
    });
  };

  const randomizeColor = () => {
    setRed(Math.floor(Math.random() * 255));
    setGreen(Math.floor(Math.random() * 255));
    setBlue(Math.floor(Math.random() * 255));
  }

  return (
    <div className='container'>
      <div className='header'>
        <h1>Цветовая палитра</h1>
        <button
          onClick={randomizeColor}>Подобрать
рандомный цвет</button>
      </div>
    </div>
  )
}
```

```

        <div className='palette-section'>
          <div className='sliders'>
            <h3>Красный</h3>
            <Slider axis="x" xstep={1} xmin={0} xmax={255}
x={red} onChange={({ x }) => setRed(x)} />
            <h3>Зелёный</h3>
            <Slider axis="x" xstep={1} xmin={0} xmax={255}
x={green} onChange={({ x }) => setGreen(x)} />
            <h3>Синий</h3>
            <Slider axis="x" xstep={1} xmin={0} xmax={255}
x={blue} onChange={({ x }) => setBlue(x)} />
            <div style={{ backgroundColor: rgbColor,
width: '200px', height: '200px', margin: '20px', border: '1px
solid lightgray'}}></div>
          </div>
          <div className='color-set'>
</div>
</div>

        <div className='color-info'>
          <h3>Выбранный вами цвет: {rgbColor}</h3>
          <button onClick={copyToClipboard}>Копировать
цвет</button>
        </div>

        <Content />
      </div>
    );
  };
};

export default ColorSlider;

```

Листинг 3.1 – Реализация кода сайта с использованием библиотеки React

Использование хуков состояния (useState) позволяет управлять текущими значениями красного, зеленого и синего компонентов цвета. Эти значения используются для формирования строки CSS rgbColor, которая отображает текущий выбранный цвет. Пользовательский интерфейс включает в себя три горизонтальных слайдера, каждый из которых соответствует одному из компонентов цвета (красный, зеленый, синий). Слайдеры позволяют пользователю точно настроить значение каждого компонента в диапазоне от 0 до 255.

Реализованы функции для изменения цвета (changeColor), копирования цвета в буфер обмена (copyToClipboard) и генерации случайного цвета (randomizeColor). Эти функции обеспечивают основную функциональность приложения.

Текущий выбранный цвет отображается в виде квадрата с фоновым цветом, соответствующим текущему значению `rgbColor`. Также предусмотрена возможность копирования этого значения в буфер обмена для дальнейшего использования.

Разработанный цветовой селектор успешно решает поставленные задачи, предоставляя пользователю удобный инструмент для выбора и работы с цветами. Использование React и его хуков состояния позволяет эффективно управлять состоянием приложения и обеспечивает высокую производительность интерфейса.