

## Домашняя работа №3: CSS-переменные и темизация сайта

**Студент:** Дмитриев Андрей Иванович

**Группа:** К3439

**Вариант:** Платформа для образовательных курсов и управления учебным процессом

---

### 1. Описание задачи

В рамках домашней работы №3 требовалось реализовать систему темизации для приложения **IT School** с использованием CSS-переменных (Custom Properties). В ходе работы необходимо было:

- выделить цветовую палитру в CSS-переменные;
  - реализовать светлую и тёмную темы;
  - добавить переключатель тем;
  - сохранять выбранную тему пользователя;
  - обеспечить плавные переходы между темами.
- 

### 2. Что такое CSS-переменные (Custom Properties)

CSS Custom Properties (CSS-переменные) — это специальные свойства CSS, позволяющие хранить и переиспользовать значения стилей. Переменные определяются с префиксом `--` и используются через функцию `var()`.

#### Преимущества CSS-переменных

- **Переиспользование** — значение задаётся один раз и применяется многократно
- **Динамическое изменение** — переменные можно менять через JavaScript
- **Каскадность** — переменные наследуются по DOM-дереву
- **Лёгкая темизация** — удобно создавать темы с переопределением значений
- **Читаемость** — семантические названия вместо набора hex-кодов

#### Синтаксис

```
:root {
```

```

--primary-color: #4A90E2;
--text-color: #1a1a1a;
--spacing-small: 8px;
}

.button {
  background-color: var(--primary-color);
  color: var(--text-color);
  padding: var(--spacing-small);
}

.element {
  color: var(--custom-color, #000000);
}

```

---

### 3. Архитектура системы темизации

#### 3.1 Структура CSS-переменных

Создан централизованный файл переменных `src/styles/themes.css`, содержащий палитру цветов, типографику, отступы, радиусы, тени, `z-index` и параметры компонентов.

```

/* =====

ГЛОБАЛЬНЫЕ CSS-ПЕРЕМЕННЫЕ

===== */

:root {

  /* — Цветовая палитра — */

  --color-primary: #4A90E2;

```

--color-primary-hover: #357ABD;

--color-primary-light: #E3F2FD;

--color-primary-dark: #1565C0;

--color-secondary: #00B18F;

--color-secondary-hover: #009977;

--color-secondary-light: #E0F7F4;

--color-accent: #FF6B6B;

--color-accent-hover: #FF5252;

--color-accent-light: #FFE5E5;

/\* Нейтральные цвета \*/

--color-background: #FFFFFF;

--color-background-secondary: #F8F9FA;

--color-background-tertiary: #F1F3F5;

--color-surface: #FFFFFF;

--color-surface-elevated: #FFFFFF;

/\* Текст \*/

--color-text-primary: #1a1a1a;

--color-text-secondary: #4a4a4a;

--color-text-tertiary: #6c757d;

--color-text-disabled: #adb5bd;

--color-text-inverse: #FFFFFF;

*/\* Границы \*/*

--color-border: #dee2e6;

--color-border-light: #e9ecef;

--color-border-dark: #adb5bd;

*/\* Состояния \*/*

--color-success: #16a34a;

--color-success-light: #dcfce7;

--color-success-dark: #15803d;

--color-error: #dc2626;

--color-error-light: #fee2e2;

--color-error-dark: #991b1b;

--color-warning: #f59e0b;

--color-warning-light: #fef3c7;

--color-warning-dark: #b45309;

--color-info: #3b82f6;

--color-info-light: #dbeafe;

--color-info-dark: #1e40af;

*/\* — Тени — \*/*

--shadow-small: 0 1px 3px rgba(0, 0, 0, 0.1);

--shadow-medium: 0 4px 6px rgba(0, 0, 0, 0.1);

--shadow-large: 0 10px 25px rgba(0, 0, 0, 0.15);

--shadow-hover: 0 6px 20px rgba(0, 0, 0, 0.15);

/\* — Отступы и размеры — \*/

--spacing-xs: 4px;

--spacing-sm: 8px;

--spacing-md: 16px;

--spacing-lg: 24px;

--spacing-xl: 32px;

--spacing-2xl: 48px;

/\* — Радиусы — \*/

--radius-sm: 4px;

--radius-md: 8px;

--radius-lg: 12px;

--radius-xl: 16px;

--radius-full: 9999px;

/\* — Типографика — \*/

--font-family-base: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',  
'Ubuntu', 'Cantarell', sans-serif;

--font-family-mono: 'Courier New', monospace;

--font-size-xs: 12px;

--font-size-sm: 14px;

--font-size-base: 16px;

--font-size-lg: 18px;

--font-size-xl: 20px;

--font-size-2xl: 24px;

--font-size-3xl: 30px;

--font-size-4xl: 36px;

--font-weight-normal: 400;

--font-weight-medium: 500;

--font-weight-semibold: 600;

--font-weight-bold: 700;

--line-height-tight: 1.25;

--line-height-normal: 1.5;

--line-height-relaxed: 1.75;

/\* — Z-index — \*/

--z-dropdown: 1000;

--z-sticky: 1020;

--z-fixed: 1030;

--z-modal-backdrop: 1040;

--z-modal: 1050;

--z-popover: 1060;

--z-tooltip: 1070;

/\* — Анимации — \*/

--transition-fast: 150ms ease;

```

--transition-base: 250ms ease;
--transition-slow: 350ms ease;

/* — Компоненты — */
--sidebar-width: 260px;
--topbar-height: 70px;
--mobile-breakpoint: 768px;

/* — Календарь — */
--schedule-accent: #00B18F;
--schedule-accent-bg: rgba(0, 177, 143, 0.08);
--schedule-accent-border: rgba(0, 177, 143, 0.55);
}

```

---

### 3.2 Тёмная тема

Для тёмной темы используется переопределение переменных через атрибут `data-theme="dark"` на корневом элементе:

```

/* =====

ТЁМНАЯ ТЕМА

===== */

[data-theme="dark"]{
  --color-primary: #5FA3E8;
  --color-primary-hover: #7BB5ED;
  --color-primary-light: #1E3A5F;
  --color-primary-dark: #3D7FC7;

```

--color-secondary: #00C9A0;  
--color-secondary-hover: #00E0B5;  
--color-secondary-light: #0A3A30;

--color-accent: #FF8585;  
--color-accent-hover: #FFA0A0;  
--color-accent-light: #4A1F1F;

--color-background: #0f0f0f;  
--color-background-secondary: #1a1a1a;  
--color-background-tertiary: #242424;

--color-surface: #1a1a1a;  
--color-surface-elevated: #242424;

--color-text-primary: #e8e8e8;  
--color-text-secondary: #b8b8b8;  
--color-text-tertiary: #888888;  
--color-text-disabled: #5a5a5a;  
--color-text-inverse: #1a1a1a;

--color-border: #333333;  
--color-border-light: #2a2a2a;  
--color-border-dark: #444444;



```
--color-success: #22c55e;
--color-success-light: #1a3a26;
--color-success-dark: #16a34a;

--color-error: #ef4444;
--color-error-light: #3a1a1a;
--color-error-dark: #dc2626;

--color-warning: #fbbf24;
--color-warning-light: #3a2f1a;
--color-warning-dark: #f59e0b;

--color-info: #60a5fa;
--color-info-light: #1a2a3a;
--color-info-dark: #3b82f6;

--shadow-small: 0 1px 3px rgba(0, 0, 0, 0.3);
--shadow-medium: 0 4px 6px rgba(0, 0, 0, 0.4);
--shadow-large: 0 10px 25px rgba(0, 0, 0, 0.5);
--shadow-hover: 0 6px 20px rgba(0, 0, 0, 0.5);

--schedule-accent: #00C9A0;
--schedule-accent-bg: rgba(0, 201, 160, 0.15);
--schedule-accent-border: rgba(0, 201, 160, 0.4);
}
```

---

### 3.3 Использование переменных в компонентах

После выноса палитры и значений в переменные все стили компонентов были переписаны на `var(...)`.

#### Пример: кнопки

```
.btn {  
  padding: var(--spacing-sm) var(--spacing-md);  
  border-radius: var(--radius-md);  
  font-size: var(--font-size-base);  
  font-weight: var(--font-weight-medium);  
  transition: all var(--transition-base);  
  border: none;  
  cursor: pointer;  
}
```

```
.btn-primary {  
  background-color: var(--color-primary);  
  color: var(--color-text-inverse);  
}
```

```
.btn-primary:hover {  
  background-color: var(--color-primary-hover);  
  box-shadow: var(--shadow-hover);  
}
```

#### Пример: карточки

```
.course-card {  
  background-color: var(--color-surface);
```

```
border: 1px solid var(--color-border);  
border-radius: var(--radius-lg);  
padding: var(--spacing-md);  
box-shadow: var(--shadow-small);  
transition: all var(--transition-base);  
}
```

```
.course-card:hover {  
  box-shadow: var(--shadow-large);  
  transform: translateY(-4px);  
}
```

### **Пример: формы**

```
.form-group input {  
  width: 100%;  
  padding: var(--spacing-md);  
  border: 2px solid var(--color-border);  
  border-radius: var(--radius-md);  
  background-color: var(--color-background);  
  color: var(--color-text-primary);  
  transition: border-color var(--transition-fast);  
}
```

```
.form-group input:focus {  
  border-color: var(--color-primary);  
  background-color: var(--color-surface);  
}
```

### Пример: навигация

```
.sidebar {  
  width: var(--sidebar-width);  
  background-color: var(--color-surface);  
  border-right: 1px solid var(--color-border);  
  z-index: var(--z-fixed);  
}
```

---

## 4. Переключатель тем

### 4.1 React-компонент ThemeToggle

Создан компонент переключения темы `src/components/ThemeToggle.jsx`, который:

- загружает сохранённую тему из `localStorage`,
- применяет её через `data-theme`,
- переключает тему по клику,
- сохраняет выбор пользователя.

```
import React, { useState, useEffect } from 'react';
```

```
import '../styles/ThemeToggle.css';
```

```
export default function ThemeToggle() {
```

```
  const [theme, setTheme] = useState('light');
```

```
  useEffect(() => {
```

```
    const savedTheme = localStorage.getItem('theme') || 'light';
```

```
    setTheme(savedTheme);
```

```
    document.documentElement.setAttribute('data-theme', savedTheme);
```

```
  }, []);
```

```

const toggleTheme = () => {
  const newTheme = theme === 'light' ? 'dark' : 'light';
  setTheme(newTheme);
  document.documentElement.setAttribute('data-theme', newTheme);
  localStorage.setItem('theme', newTheme);
};

return (
  <button
    onClick={toggleTheme}
    className="theme-toggle"
    aria-label={`Переключить на ${theme === 'light' ? 'тёмную' : 'светлую'} тему`}
    title={`Текущая тема: ${theme === 'light' ? 'светлая' : 'тёмная'}`}
  >
    <span className="theme-toggle-icon">
      {theme === 'light' ? '🌙' : '☀️'}
    </span>
    <span className="theme-toggle-text">
      {theme === 'light' ? 'Тёмная' : 'Светлая'}
    </span>
  </button>
);
}

```

---

## 4.2 Стили ThemeToggle

```
.theme-toggle {  
  display: flex;  
  align-items: center;  
  gap: var(--spacing-sm);  
  padding: var(--spacing-sm) var(--spacing-md);  
  background-color: var(--color-background-secondary);  
  border: 1px solid var(--color-border);  
  border-radius: var(--radius-full);  
  cursor: pointer;  
  transition: all var(--transition-base);  
  font-size: var(--font-size-sm);  
  color: var(--color-text-primary);  
}
```

```
.theme-toggle:hover {  
  background-color: var(--color-background-tertiary);  
  border-color: var(--color-primary);  
  box-shadow: var(--shadow-small);  
}
```

---

### 4.3 Интеграция в TopBar

```
import ThemeToggle from './ThemeToggle';
```

```
export default function SmartTopBar() {  
  return (  
    <header className="topbar">
```

```
<div className="topbar-right">

  <ThemeToggle />

  <NotificationBell />

  <UserMenu />

</div>

</header>

);
}
```

---

## 5. Плавные переходы между темами

Для плавного визуального переключения добавлены переходы на основные свойства:

```
* {
  transition:
    background-color var(--transition-base),
    border-color var(--transition-base),
    color var(--transition-base),
    box-shadow var(--transition-base);
}

@media (prefers-reduced-motion: reduce) {
  * {
    transition: none !important;
    animation: none !important;
  }
}
```

---

## 6. Сохранение выбора темы

### 6.1 LocalStorage

```
localStorage.setItem('theme', 'dark');  
  
const savedTheme = localStorage.getItem('theme') || 'light';
```

### 6.2 Инициализация темы при старте приложения

```
useEffect(() => {  
  
  const savedTheme = localStorage.getItem('theme') || 'light';  
  
  document.documentElement.setAttribute('data-theme', savedTheme);  
  
}, []);
```

---

## 7. Примеры применения в компонентах

Стили главной страницы, карточек курсов, модальных окон, таблиц и toast-уведомлений были переписаны на CSS-переменные. Это обеспечило консистентность дизайна и корректную работу обеих тем без дублирования CSS.

---

## 8. Адаптация сторонних библиотек

### 8.1 FullCalendar

Переопределены CSS-переменные FullCalendar для согласования с темами приложения:

```
.fc {  
  
  --fc-border-color: var(--color-border);  
  
  --fc-button-bg-color: var(--color-primary);  
  
  --fc-button-hover-bg-color: var(--color-primary-hover);  
  
  --fc-today-bg-color: var(--color-primary-light);  
  
}
```

### 8.2 React Toastify



Переопределены переменные и классы Toastify, чтобы уведомления соответствовали текущей теме.

---

## **9. Тестирование темизации**

### **9.1 Проверочный список**

- Все основные страницы адаптированы под темы
- Переход между темами плавный
- Тема сохраняется в localStorage
- Тема восстанавливается при перезагрузке
- Контраст в обеих темах соответствует требованиям
- Переключатель имеет aria-label
- Поддерживается prefers-reduced-motion

### **9.2 Кроссбраузерность**

Проверено в: Chrome, Firefox, Safari, Edge — CSS-переменные корректно работают во всех современных браузерах.

---

## **10. Преимущества реализованной системы**

### **Для разработки**

- централизованное управление палитрой и токенами;
- простое добавление новых тем;
- меньше дублирования стилей;
- единая дизайн-система.

### **Для пользователя**

- возможность выбрать комфортную тему;
  - сохранение выбора между сессиями;
  - плавные переходы без резких изменений.
-

## 11. Дополнительные возможности

Система допускает расширение: авто-тема по времени, режим “high-contrast”, дополнительные варианты тем (warm/cool) — за счёт добавления новых блоков [data-theme=“...”].

---

## 12. Скриншоты

В отчёт добавлены скриншоты:

- светлая тема (главная, курсы);
  - тёмная тема (главная, курсы);
  - переключатель темы;
  - анимация перехода.
- 

## 13. Структура файлов

src/

```
├─ styles/
|   ├─ themes.css
|   ├─ ThemeToggle.css
|   ├─ HomePage.css
|   ├─ LoginPage.css
|   ├─ CourseCard.css
|   ├─ Sidebar.css
|   └─ Modal.css
└─ ...
├─ components/
|   └─ ThemeToggle.jsx
└─ ...
```

## 14. Выводы

В ходе выполнения домашней работы №3 реализована полноценная темизация приложения **IT School** на основе CSS-переменных:

- создана централизованная дизайн-система токенов (цвета, типографика, размеры, тени);
- реализованы **светлая** и **тёмная** темы через переопределение переменных;
- добавлен переключатель темы, сохраняющий выбор в **localStorage**;
- добавлены плавные переходы и поддержка prefers-reduced-motion;
- стили компонентов и сторонних библиотек адаптированы к тематизации.

Темизация улучшает UX, делает интерфейс современным и масштабируемым, а также упрощает поддержку и развитие проекта.

**Дата выполнения:** 21.01.2026