

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ
УНИВЕРСИТЕТ ИТМО

Факультет инфокоммуникационных технологий

Образовательная программа 09.03.02

Направление подготовки (специальность) Мобильные сетевые технологии

О Т Ч Е Т

по практической работе 3

Тема задания: CSS-переменные, темизация сайта средствами CSS

Обучающийся Сулейманов Руслан Имранович, К33402

Преподаватель: Добряков Д. И., преподаватель

Санкт-Петербург

Задача: выполнить темизацию ранее реализованного сайта

Ход работы: Реализовано три кнопки, отвечающие за темную, светлую темы и активную тему компьютера пользователя.



Для этого было создано 2 файла переменных для светлой и темной теме:
dark.css

```
:root {
  --back-color: #273a43;
  --back-image: url('../dark.jpg');
  --body-color: #ffffff0;
  --body-shadow: rgba(255, 160, 122, 0.6);
  --body-glow: rgba(255, 255, 255, 0.7);
  --a-color: #fff;
  --text-color: #fff;
  --a-hover-color: #848484;
  --a-active-color: #848484;
  --a-bg-color: #00070a;
  --body-item-color: #fff;
}
```

light.css

```
1  :root {
2  --back-color: #fff;
3  --back-image: url('../light.jpg');
4  --body-color: #557f95;
5  --body-shadow: rgba(222, 168, 33, 0.4);
6  --body-glow: rgba(70, 130, 180, 0.4);
7  --a-color: #7957d5;
8  --text-color: #000;
9  --a-hover-color: #020000;
10 --a-active-color: #020000;
11 --a-bg-color: #fff;
12 --body-item-color: #000;
13 }
```

Main.css

```

:root {
  color-scheme: light dark;
}

html, body {

```

В файле которого мы используем активный файл стилей.
Подключены стили в index.html:

```

<meta name="color-scheme" content="light dark">
<script>
  // If 'prefers-color-scheme' is not supported, fall back to light mode.
  // In this case, light.css will be downloaded with 'highest' priority.
  if (window.matchMedia( query: '(prefers-color-scheme: dark)').media === 'not all') {
    document.documentElement.style.display = 'none';
    document.head.insertAdjacentHTML(
      where: 'beforeend',
      html: '<link rel="stylesheet" href="./css/light.css" onload=document.documentElement.style.display = \'\'>',
    );
  }
</script>
<link rel="stylesheet" href="css/light.css" media="(prefers-color-scheme: light)">
<link rel="stylesheet" href="css/dark.css" media="(prefers-color-scheme: dark)">
<link rel="stylesheet" href="css/main.css">

```

Где включена проверка на наличие системной темы у пользователя.

Также реализован стор, для хранения темы в локалсторе:

```

changeTheme (state, payload :{theme: string} ) {
  state.theme = payload.theme;
  localStorage.setItem('theme', payload.theme)
  if (state.theme==='auto') {
    state.activeTheme = window.matchMedia && window.matchMedia( query: '(prefers-color-scheme: dark)').matches? 'dark':'light';
  }

  if (state) {
    let lightMedia;
    let darkMedia;

    if (state.theme === 'auto') {
      lightMedia = '(prefers-color-scheme: light)';
      darkMedia = '(prefers-color-scheme: dark)';
    } else {
      lightMedia = (state.theme === 'light') ? 'all' : 'not all';
      darkMedia = (state.theme === 'dark') ? 'all' : 'not all';
    }

    [...lightStyles].forEach((link) => {
      link.media = lightMedia;
    });

    [...darkStyles].forEach((link) => {
      link.media = darkMedia;
    });
  }
}

```

Функция смены темы, меняет значение в локалсторе, и в стейте
Кнопки смены темы:

```

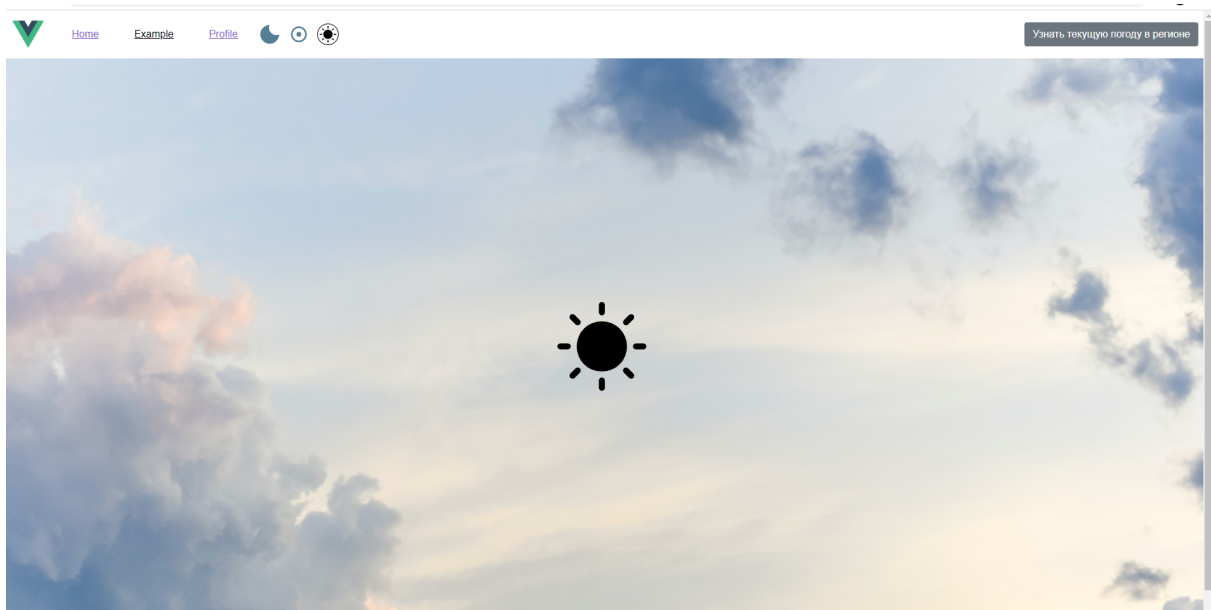
<button v-bind:class="{ active: $store.state.theme.theme==='dark' }" class="theme-btn" @click="$store.commit('changeTheme', {theme: '
  <svg>
    <use xlink:href="#moon"></use>
  </svg>
</button>
<button v-bind:class="{ active: $store.state.theme.theme==='auto' }" class="theme-btn" @click="$store.commit('changeTheme', {theme: 'auto'})">
  <svg>
    <use xlink:href="#auto"></use>
  </svg>
</button>
<button v-bind:class="{ active: $store.state.theme.theme==='light' }" class="theme-btn" @click="$store.commit('changeTheme', {theme: 'light'})">
  <svg>
    <use xlink:href="#sun"></use>
  </svg>

```

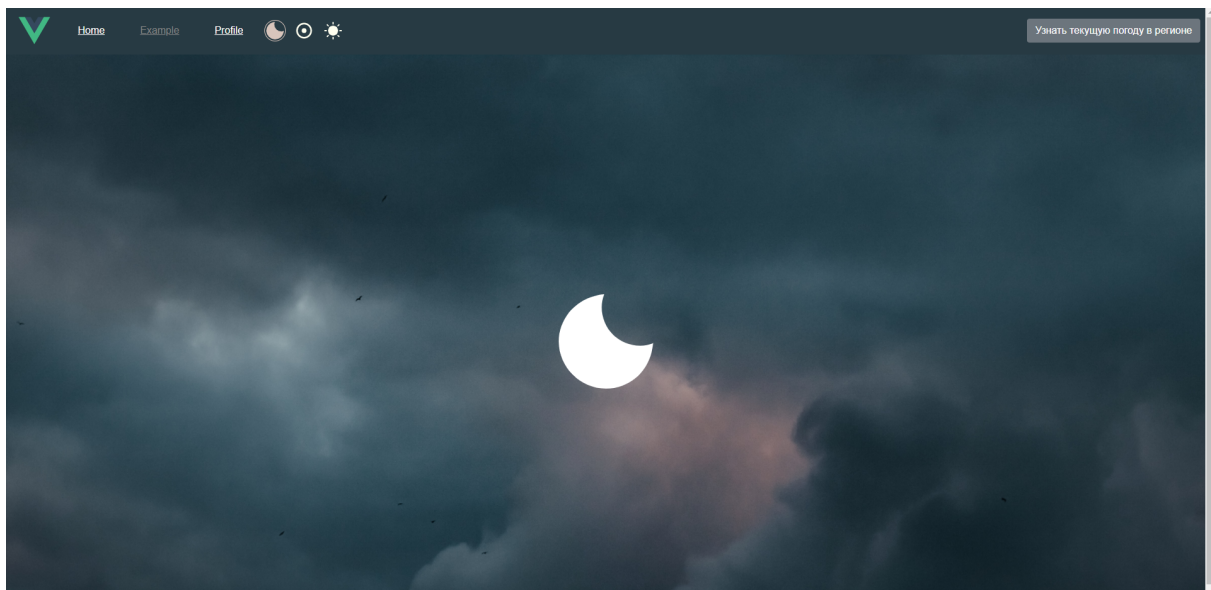
(Вся логика описана в темплейтах, так как первые пробы во vue)

Результат:

Светлая тема:



Темная тема:



Как можно заметить, меняется как само содержимое страницы, так и хедер данной страницы, подстраивается под темную тему.

Пример использования переменных в коде файла:

```
#theme_page #body {  
  width: 20vmin;  
  height: 20vmin;  
  border-radius: 50%;  
  background-image: radial-gradient(  
    circle at 20% 20%,  
    var(--body-color) 50%,  
    var(--body-shadow)  
  );  
  box-shadow: 0 0 10vmin var(--body-glow);  
}  
  
.navbar {  
  background-color: var(--back-color)!important;  
}
```

Выводы: реализован свитчер темы сайта, с сохранением его в локал сторе. Изучен метод построения логики для реализации данного функционала.