

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Фронт-энд разработка

**Отчет**

**Домашняя работа №3**

**Выполнила:**

**Кадникова Екатерина**

**Группа К3441**

**Проверил:  
Добряков Д. И.**

**Санкт-Петербург**

**2025 г.**

## **Цель задания**

Цель - выполнить темизацию ранее реализованного сайта. Добавить к текущему варианту сайта дополнительную тему, в итоге должно получиться либо: светлая и тёмная с ориентиром на пользовательские настройки. Либо две кастомные темы с переключателем через JS.

## **Подход и структура реализации**

### **CSS-переменные**

Для удобства темизации были определены CSS-переменные, которые описывают основные цвета и элементы интерфейса.

Пример переменных:

```
:root {  
    --bg-body: #f8f9fa;  
    --color-text: #333;  
    --color-text-muted: #6c757d;  
  
    --primary: #0d6efd;  
    --primary-dark: #0056b3;  
    --card-bg: #fff;  
    --card-text: #212529;  
    --input-bg: #fff;  
    --input-color: #212529;  
}
```

Для тёмной темы был создан отдельный класс `.dark-theme`, где те же переменные переопределяются:

```
.dark-theme {  
    --bg-body: #1c1e23;  
    --color-text: #e0e0e0;  
    --color-text-muted: #a0a0b0;  
    --primary: #3390ff;  
    --card-bg: #2a2d34;  
    --card-text: #e0e0e0;  
    --input-bg: #2a2d34;
```

```
--input-color: #e0e0e0;  
}
```

Такой подход позволяет изменять цвета всего сайта одной строчкой кода, просто переключая класс dark-themeey body.

### Общие стили с использованием переменных

Пример стилей для карточек и текста:

```
.card, .auth-card, .property-card {  
    background-color: var(--card-bg);  
    color: var(--card-text);  
    border: 1px solid var(--border-color);  
    border-radius: 0.5rem;  
}
```

```
.text-muted {  
    color: var(--color-text-muted) !important;  
}
```

```
small {  
    color: var(--color-text-muted);  
}
```

Использование var(--...) позволяет автоматически адаптироваться к выбранной теме. Префикс !important для text-muted обеспечил переопределение любых встроенных стилей Bootstrap.

### Стили форм регистрации и авторизации

Для форм была сделана поддержка обоих режимов:

```
.form-control, .form-select {  
    background-color: var(--input-bg);  
    color: var(--input-color);  
    border: 1px solid var(--input-border);  
}
```

```
.form-control:focus, .form-select:focus {  
    border-color: var(--input-focus-border);  
    box-shadow: 0 0 0 0.2rem var(--input-focus-shadow);  
}
```

Пример на регистрации: текст под заголовком формы тоже использует text-muted:

```
<p class="text-muted text-center">Создайте новый  
аккаунт</p>
```

В тёмной теме этот текст меняет цвет на более светлый автоматически через CSS-переменные.

### Переключатель темы

Реализован как кнопка с текстом и ползунком, с сохранением состояния:

```
const themeToggleBtn = document.createElement('button');  
themeToggleBtn.style.width = '180px';  
themeToggleBtn.style.height = '50px';  
themeToggleBtn.style.borderRadius = '25px';  
themeToggleBtn.style.display = 'flex';  
themeToggleBtn.style.alignItems = 'center';  
themeToggleBtn.style.justifyContent = 'flex-start';  
themeToggleBtn.style.backgroundColor = '#0d6efd';  
themeToggleBtn.style.color = '#fff';  
  
const toggleText = document.createElement('span');  
toggleText.innerText = localStorage.getItem('theme') ===  
'dark' ? 'Светлая тема' : 'Темная тема';  
toggleText.style.flexGrow = '1';  
toggleText.style.textAlign = 'center';  
themeToggleBtn.appendChild(toggleText);  
  
const toggleCircle = document.createElement('div');  
toggleCircle.style.width = '36px';
```

```
toggleCircle.style.height = '36px';
toggleCircle.style.borderRadius = '50%';
toggleCircle.style.backgroundColor = '#fff';
toggleCircle.style.marginLeft = '5px';
toggleCircle.style.transform =
localStorage.getItem('theme') === 'dark' ?
'translateX(130px)' : 'translateX(0)';
themeToggleBtn.appendChild(toggleCircle);

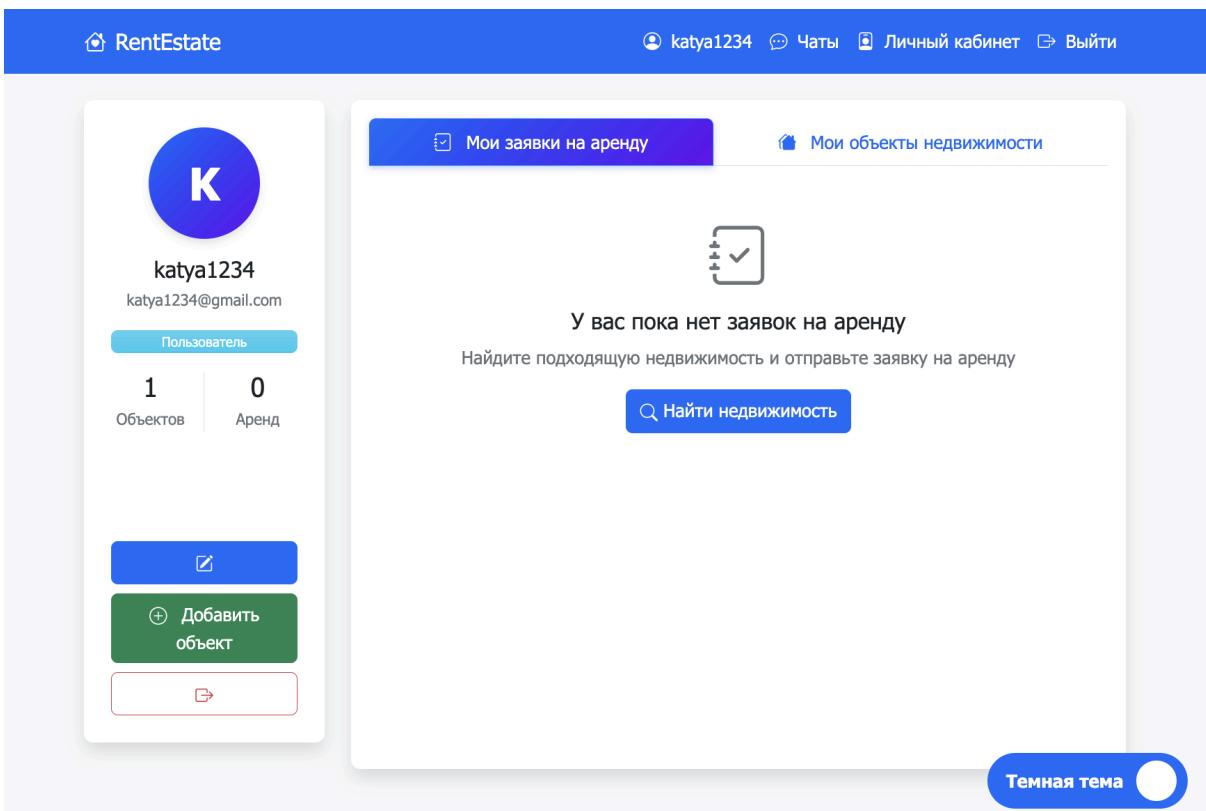
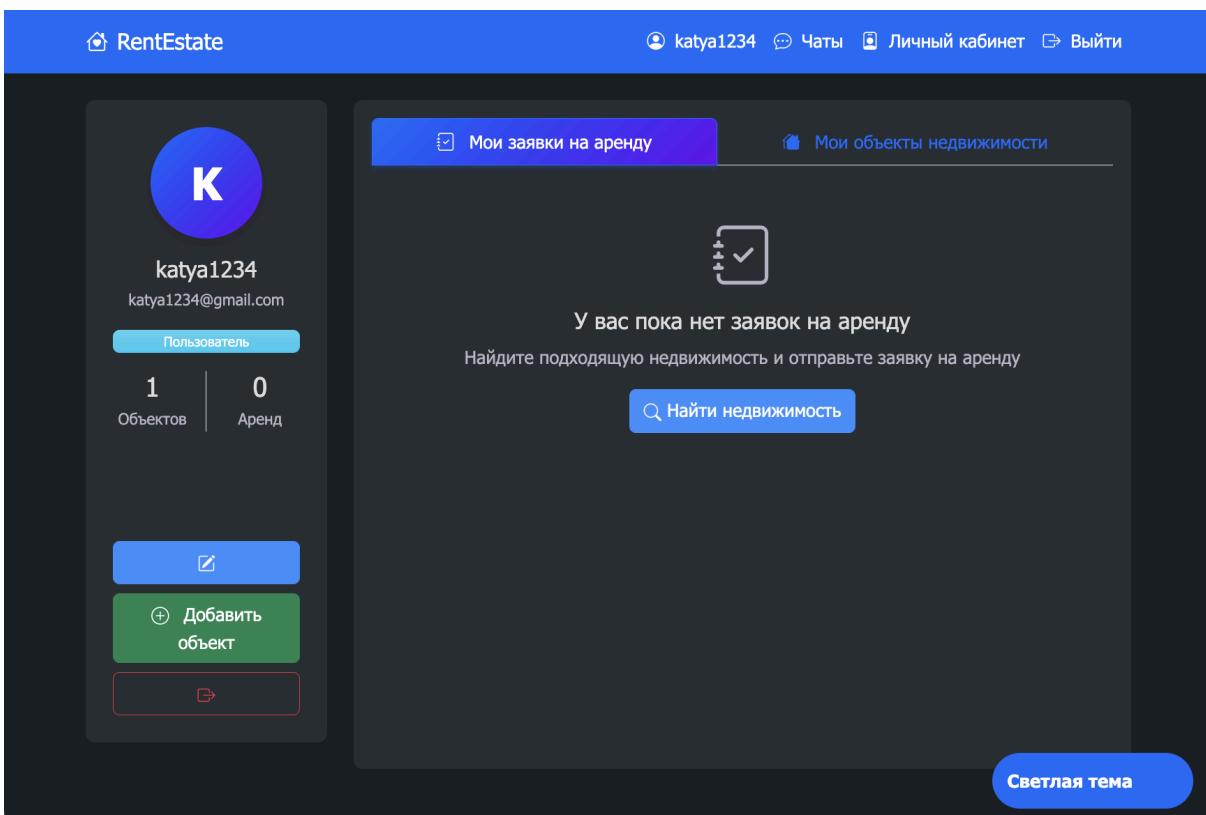
themeToggleBtn.addEventListener('click', () => {
    document.body.classList.toggle('dark-theme');
    const isDark =
document.body.classList.contains('dark-theme');
    toggleText.innerText = isDark ? 'Светлая тема' :
'Tемная тема';
    toggleCircle.style.transform = isDark ?
'translateX(130px)' : 'translateX(0)';
    localStorage.setItem('theme', isDark ? 'dark' :
'light');
});
```

Текст внутри кнопки объясняет пользователю назначение переключателя. Кнопка адаптируется к теме, изменяя цвет фона и положение ползунка. Состояние сохраняется в localStorage, поэтому при перезагрузке сайта сохраняется выбранная тема.

### Примеры обновленного отображения элементов

Карточка профиля:

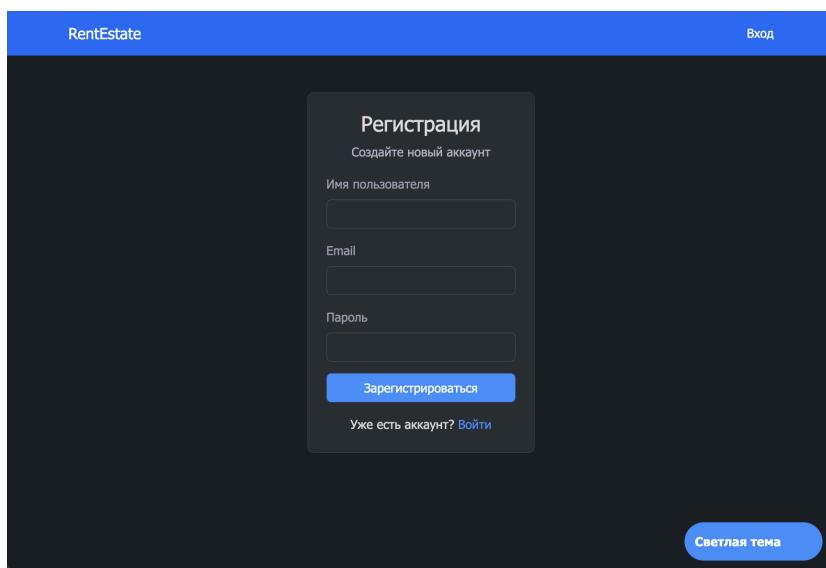
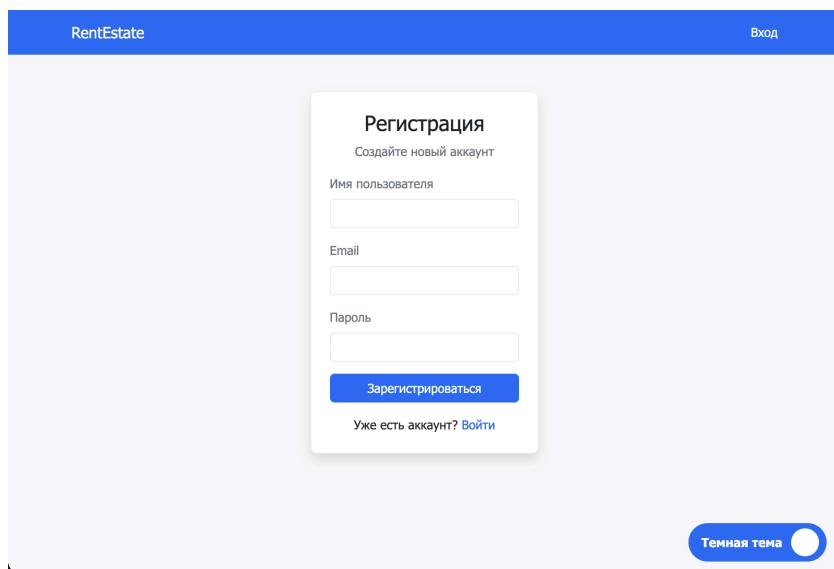
```
<div class="col-6 border-end">
    <div class="h4 mb-1" id="propertiesCount">0</div>
    <div class="text-muted small">Объектов</div>
</div>
```



Форма регистрации:

```
<p class="text-muted text-center">Создайте новый
```

аккаунт</p>



## Выводы

Созданы CSS-переменные для двух тем. Реализован переключатель темы с визуальной индикацией и сохранением состояния. Все элементы сайта корректно реагируют на смену темы.