

УПРАВЛЕНИЕ ИОТ УСТРОЙСТВАМИ С ОТКРЫТЫМ API ЧЕРЕЗ WEB-ПРИЛОЖЕНИЕ

М. А. Краснов

*Белорусский государственный университет,
пр. Независимости, 4, 220030, г. Минск, Беларусь, grektop502@gmail.com
Научный руководитель: А. А. Дерюшев, кандидат технических наук, доцент*

В работе представлена разработка web-приложения для управления умными устройствами в локальной сети. Основным результатом является приложение, позволяющее управлять умными устройствами, на примере взаимодействия с Yeelight smart LED bulb w3 (multicolor).

Ключевые слова: IoT; веб-приложение; лампочка LED smart bulb; интернет вещей.

Введение

В эпоху цифровизации и стремительного технологического прогресса, умные устройства, или устройства Интернета вещей (IoT), стали неотъемлемой частью нашей повседневной жизни. Они проникают в различные сферы нашего существования, включая домашнюю обстановку, промышленность, здравоохранение и городскую инфраструктуру.

Самыми распространенными примерами применения технологии IoT являются умные дома и умные города. Умные дома используют IoT устройства для автоматизации и оптимизации бытовых процессов, в то время как умные города применяют их для повышения эффективности городских служб и улучшения качества жизни горожан.

Однако управление этими устройствами может быть сложным, особенно в случае открытых API. В связи с этим возникает необходимость в разработке веб-приложений, которые могут упростить процесс управления и координации этими устройствами. В рамках данного исследования будет изучаться и разрабатываться методика управления устройствами IoT с открытыми API через веб-приложение.

Цель работы

Основной целью данной работы является разработка web-приложения, построенного согласно паттерну Клиент-Сервер, для управления умной лампочкой Yeelight smart LED bulb w3, находящейся в той же локальной сети что и сервер web-приложения. Существует оригинальное приложение для мобильных устройств, позволяющее управлять умной лампочкой, но у него есть свои недостатки: в приложении нет возможности устанавливать цвет лампочки, используя точные значения одной из палитр (RGB, HSV), у многих пользователей возникают сложности с подключением лампочки к приложению. Разрабатываемое приложение призвано решить эти проблемы.

Основные функции приложения:

1. Включение/выключение умного устройства;
2. Изменение цвета лампочки с использованием RGB/HSV схемы;
3. Добавление нового расписания, в используемое устройство;
4. Поиск доступных устройств в локальной сети.

Задачи работы

1. Изучение API-документации для устройства Yeelight smart LED bulb w3: получение основных правил/методов для управления данным устройством через внешнее API.

2. Разработка серверной части приложения на Node.js с использованием библиотеки Express.js: написание основных функций, роутов, необходимых для взаимодействия с устройством.

3. Разработка Front-End части приложения с использованием фреймворка React: создание основных страниц, позволяющих удобно взаимодействовать с сервером и видеть, что происходит с устройством.

4. Анализ результатов и выводы.

Изучение API-документации устройства

В начале работы проводились подробное изучение документации с дальнейшим выделение основных, необходимых для работы методов взаимодействия с устройством, анализ кода, предоставляемого сторонними разработчиками, а также изучение списка библиотек для возможного использования. Основной задачей являлся сбор основных методов, библиотек для успешной реализации приложения. Был выбран модуль `net`, предоставляемый Node.js, позволяющий устанавливать соединение с устройством, а также модуль `dgram` для поиска устройства в локальной сети.

Разработка серверной части приложения

Следующим шагом было написание серверной части приложения. Как упоминалось ранее, для этого были использованы: библиотека Express.js, модули `net` и `dgram`. Данные модули были выбраны по нескольким причинам:

1. Модуль `net` позволяет устанавливать соединение с лампочкой, посредством сокетов;
2. Модуль `net` прост и интуитивно понятен для использования;
3. Модуль `dgram` позволяет отправлять UDP запросы устройствам, находящимся в локальной сети, что существенно упростит поиск информации о лампочке для дальнейшего взаимодействия с ней;
4. Библиотека Express.js позволяет не беспокоиться об используемых сетевых протоколах и низкоуровневых процессах, что упрощает разработку.

Разработка Front-End части приложения

В ходе следующего шага были созданы web-страницы для приложения с использованием фреймворка React. Всего было разработано 2 страницы. Первая отвечала за поиск устройства в локальной сети, а вторая за отображения основной информации об устройстве, а также содержащая интерфейсы для взаимодействия с устройством (кнопки, поля ввода, ползунки-диапазона). Кнопки предназначены для включения/выключения лампочки, поля ввода отвечают за установку времени для расписания, а ролью ползунков является изменение цвета лампочки согласно цветовым схемам RGB/HSV.

Анализ результатов и выводы

Завершив процесс создания приложения, стоит отметить, что при наличии необходимых устройств, а также среднего уровня знаний языка js, библиотеки Express.js и некоторых ее модулей, того, как работают приложения, построенные согласно паттерну Клиент-Сервер, становится возможным создание своего приложения для управления умными устройствами с открытым API.

Результаты работы

1. Изучена API-документация для взаимодействия с умной лампочкой.
2. Разработана серверная часть данного web-приложения.
3. Реализована Front-End часть приложения.
4. Результаты работы были проанализированы и сделаны соответствующие выводы.

Перспективы развития

Основной перспективой развития данного приложения является добавления возможность взаимодействия с другими умными устройствами, а также добавление возможности для удаленного управления IoT устройствами, то есть не находящимися в локальной сети.

Библиографические ссылки

1. Yeelight WiFi Light Inter-Operation Specification [Electronic resource] / ed. Qingdao Yeelink Information Technology Co., Ltd.
URL: https://home.yeelight.de/site/templates/downloads/yeelight_inter-operation-spec.pdf (date of access: 22.03.2024).
2. ExpressJS Tutorial Absolute Learning [Electronic resource] / ed. Tutorials point.
URL: <https://www.tutorialspoint.com/expressjs/index.htm> (date of access: 22.03.2024).
3. Node.js v21.7.3 documentation [Electronic resource] / ed. TJ Holowaychuk.
URL: <https://nodejs.org/api/dgram.html> (date of access: 22.03.2024).