

УДК 004.04

БАЗОВЫЕ ТРЕБОВАНИЯ К СИСТЕМЕ МОНИТОРИНГА УСТРОЙСТВ ИНТЕРНЕТА ВЕЩЕЙ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ

Смыков И.М., Полторак А.В., Карусевич Т.Е.

*МИРЭА — Российский технологический университет» 119454, Россия, г. Москва, проспект Вернадского, 78,
e-mail: ivansmykoff@yandex.ru, alpo11@yandex.ru, tatyana1427@yandex.ru*

Количество устройств, имеющих доступ в интернет растет. Большое количество систем, связанных в сети интернет с устройствами получения данных (устройствами Интернета вещей), способны передавать и обрабатывать данные от данных устройств. Одной из востребованных функций указанных систем является мониторинг устройств Интернета вещей в режиме реального времени. Выделены базовые требования. Построена архитектура системы управления и мониторинга производительности интернета вещей.

Ключевые слова: мониторинг, устройства Интернета вещей, режим реального времени, система, компоненты программы.

BASIC REQUIREMENTS FOR MONITORING DEVICES OF INTERNET OF THINGS IN REAL TIME

Smykov I.M., Poltorak A.V., Karusevich T.E.

*MIREA - Russian Technological University, 119454, Moscow, 78 Vernadskogo Avenue, Russia,
e-mail: ivansmykoff@yandex.ru, alpo11@yandex.ru, tatyana1427@yandex.ru*

The number of devices with Internet access is growing. A large number of systems connected on the Internet with data acquisition devices (IoT devices) are capable of transmitting and processing data from these devices. One of the demanded functions of these systems is real-time monitoring of IoT devices. The basic requirements are highlighted. The architecture of the IoT performance management and monitoring system has been built.

Key words: monitoring, IoT devices, real-time mode, system, program components.

Введение

Мониторинг – это система постоянного наблюдения за явлениями и процессами, проходящими в окружающей среде, результаты которого служат для обоснований принимаемых решений в сфере, где используются данные, мониторинг которых осуществляется.

При создании таких систем, их архитекторы и разработчики сталкиваются с похожими проблемами. Имеются в виду проблемы на пути к достижению масштабности, производительности, доступности системы.

В результате анализа аналогов программных систем мониторинга устройств IoT в режиме реального времени обнаружено, что имеющимся на рынке продуктам присущий недостаток в виде отсутствия формата сериализации кроме JSON, который хоть и решает задачу сериализации данных, но является довольно объемным. Поэтому предлагается создать программную систему, которая могла бы в качестве формата сериализации полезной нагрузки направляемых сообщений использовать более сжатые форматы сериализации, такие как, например MessagePack или Google Protobuf, которые являются более эффективными, чем JSON [3].

Поскольку в рассматриваемых программных средствах не поддерживаются эти форматы сериализации, то в них имеется проблема минимизации использования интернет трафика. Разрабатываемое ПО призвано решить проблему использования данных, направляемых данным Интернету вещей [1,5].

Требования к системе мониторинга устройств интернета вещей в режиме реального времени

При анализе предметной области систем мониторинга Интернета вещей, были определены следующие функциональные характеристики разрабатываемого программного продукта:

- Аутентифицированные пользователи должны иметь возможность регистрировать новые устройства в системе и в результате получать уникальные идентификаторы для них. На этом этапе нужно предусмотреть валидацию входных данных от пользователя.

- Должна быть предусмотрена возможность удалить зарегистрированное в системе устройство.
- Должна быть возможность модифицировать данные об уже существующем устройстве, за исключением уникально присвоенного идентификатора, его изменение может быть выполнено только повторной генерацией уникального значения.
- Пользователь должен иметь возможность просматривать сохраненные данные, направленные от устройств, которые он зарегистрировал.
- Пользователи должны иметь возможность получать данные зарегистрированных ими устройств Интернета вещей в реальном времени.
- Потенциальные пользователи должны иметь возможность зарегистрироваться в системе, введя свои данные.
- Уже зарегистрированные пользователи должны иметь возможность войти в систему, введя свой логин и пароль.
- Зарегистрированные и подключенные к системе устройства должны иметь возможность отправлять данные об измерениях по предварительно согласованным протоколам.
- Обмен данными между системой и устройствами должен осуществляться с использованием HTTP протокола в качестве транспортного.
- Все сообщения, отправленные устройствами, должны быть сериализованы в один из предварительно согласованных форматов.
- Система должна успешно поддерживать такие форматы сериализации, как JSON, Protocol Buffers, MessagePack.

Рекомендуемые системные требования к данной системе не могут быть определены точно, но возможно предоставить рекомендации по повышению производительности работы веб-приложения.

Первой из рекомендаций является использование современных версий веб-браузеров, поскольку в них присутствует поддержка технологии веб-сокетов, которая является наиболее надежной и быстрой и может быть использована для взаимодействия в реальном времени. Также новые версии браузеров увеличивают свою производительность с каждой версией, что является еще одной причиной для их использования. Второй рекомендацией является использование большего количества оперативной памяти и мощного процессора.

Клиентская машина должна иметь графический интерфейс и иметь достаточные минимальные характеристики, чтобы запускать такие браузеры как: Internet Explorer 8,9,10,11; MS Edge; Google Chrome; Opera; Safari; Android Browser.

Архитектура программной системы мониторинга

В результате анализа проблемы была разработана высокоуровневая архитектура программной системы, которая представлена на рис. 1.

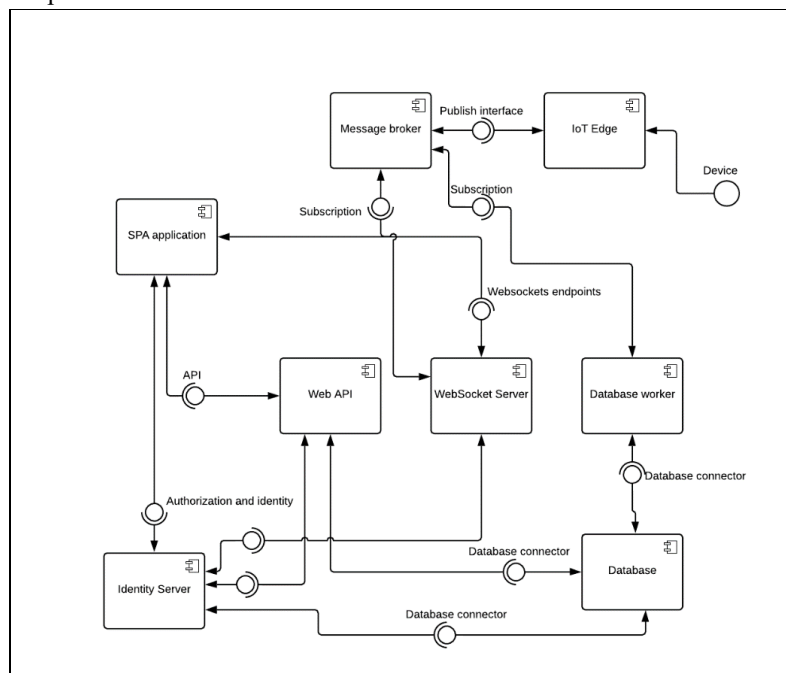


Рис.1. Диаграмма компонентов программной системы мониторинга устройств IoT в режиме реального времени

Преимуществом данной архитектуры системы являются:

- масштабируемость - поскольку каждый компонент является отдельной частью с отдельной ответственностью, то при увеличении нагрузки на какую-то часть программной системы каждый компонент может быть горизонтально масштабированным отдельно от других;
- Слабая связанность - при сбое одного из компонентов вся система продолжает работать в ожидании возобновления работы компонента, который вышел из строя;
- Параллельность разработки системы - при данной архитектуре компоненты могут быть поделены между командами, и их разработка может вестись параллельно.

Модель данных представлена на рис.2.

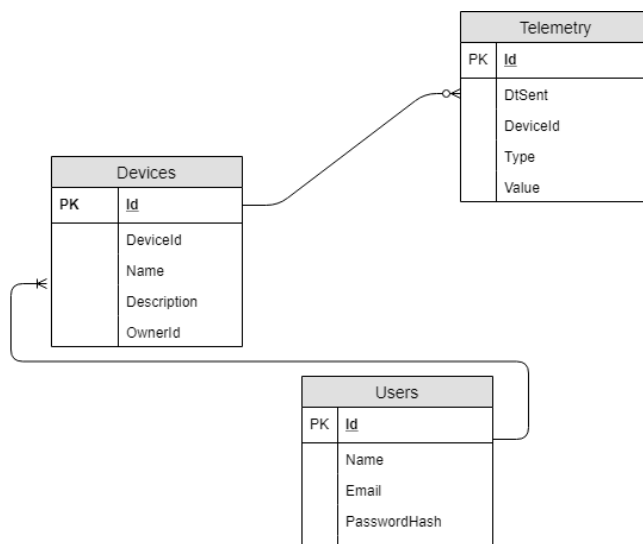


Рис.2.Модель данных

Рассмотрим основные компоненты системы.

1. «Device». Этот компонент представляет конкретное подключенное к системе устройство Интернета вещей. На рисунке 2 показан как таблица Devices и таблица имеет следующие атрибуты:

- Id – уникальный номер устройства в таблице и предназначается для идентификации внутри программной системы. Представляется целочисленным видом данных и является первичным ключом в таблице.
- DeviceId – уникальный номер устройства, предназначается не только для его использования внутри системы, но и для его использования за его пределами конечными пользователями, то есть, как и устройствами, так и людьми. Представлен типом данных Guid (Global Unique Identifier), представляющий собой идентификаторы, генерируемые с высокой степенью уникальности.
- Name – имя устройства, данное ему пользователем. Не является уникальным полем и возможны случаи, когда несколько разных пользователей имеют одинаковые названия устройств. Представляется строчным типом данных.
- Description – описание устройства, опциональное поле, служит для описания устройства пользователя.
- OwnerId – идентификатор пользователя, которому принадлежит данное устройство. Представляется целочисленным типом данных.

2. Компонент «Users». Представляет собой пользователя системы и имеет следующие атрибуты:

- Id – уникальный Id пользователя в системе. Представлено целочисленным типом данных и является первичным ключом в таблице.
- Name – уникальное имя пользователя в системе. Представлено строчным типом данных.
- Email – электронная почта пользователя. Представлено строчным типом данных.
- PasswordHash – пароль пользователя, предварительно захеширован надежным алгоритмом хэширования (алгоритм является деталью реализации компонента Identity Server). Хэш пароля сохраняется для того, чтобы избежать возможности кражи паролей пользователей (злоумышленник, имея хэш пароля, не сможет его восстановить, поскольку это основное свойство хэш функций). Представлено строчным типом данных.

3. Компонент «Telemetry». Представляет собой результат направленного измерения конкретного устройства

в конкретный момент времени. Содержит следующие атрибуты:

- Id - уникальный идентификатор измерения в таблице, является первичным ключом в таблице и представлен целочисленным типом данных.
- DtSent - дата и время, когда устройством было осуществлено данное измерение. Сохраняется в формате данных, представляющий даты.
- DeviceId - идентификатор устройства, которое прислало данные, сохраняется как уникальный идентификатор (Guid).
- Type - тип измерения, например: температура, влажность, освещенность - сохраняется как целочисленное значение.
- Value - значение измерения, сохраняется как число с плавающей точкой.

Для тестирования разработанного программного средства были выбраны следующие виды тестирования:

- По запуску кода на тестирование - статическое тестирование.
- По доступу к коду и архитектуре приложения - метод черного ящика. Тестирование методом черного ящика осуществляется без доступа к исходному коду программы.
- По уровню детализации было выбрано провести такие виды тестирования, как интеграционное и модульное.
- По принципам работы с приложением – положительное тестирование. Все действия с приложением выполняются строго по инструкции без всяких допустимых действий и некорректных данных.

Описание системы мониторинга

Выполним описание разрабатываемой системы. Запуская браузер и переходя по адресу, где развернуто программное средство, пользователь имеет возможность наблюдать страницу для входа в систему.

Для того, что войти в систему, пользователь должен ввести свой адрес электронной почты и пароль, а затем нажать кнопку «Login». После удачного входа в систему пользователь может видеть устройства, зарегистрированные им. Каждое устройство имеет краткое описание с названием и кнопкой «View details».

Нажав данную кнопку, пользователь имеет возможность перейти на подробное описание конкретного устройства IoT. Сверху отображается кнопка для регистрации нового устройства, нажав которую пользователь переходит на страницу для регистрации нового устройства.

Для того, чтобы зарегистрировать новое устройство, пользователь должен ввести имя устройства и также добавить к нему опциональное описание. Пока пользователь не введет имя устройства, кнопка «Register device» не будет подсвечиваться и таким образом будет невозможно зарегистрировать новое устройство без имени.

Для того, чтобы вернуться на предыдущую страницу, нужно нажать на кнопку «Go back to devices», что осуществит навигацию на предыдущую страницу со списком всех устройств.

Для того, чтобы попасть на страницу с описанием конкретного выбранного устройства, на странице со всеми устройствами нужно выбрать нужное и нажать клавишу «View details».

На этой странице, графический интерфейс которой представлен на рис. 3, пользователь сможет просмотреть все данные устройства, включая автоматически сгенерированные идентификатором, который будет использоваться для его уникальной идентификации в системе.

Также доступны такие атрибуты устройства, как его имя и описание. Для того, чтобы удалить устройство из системы предусмотрена кнопка Delete, которая удалит устройство из системы навсегда.

На рис.3 представлена секция «Received measurements» на которой пользователь может видеть график, который обновляется в реальном времени.

Также на этой странице присутствует кнопка «Go back to devices», которая осуществляет навигацию в список зарегистрированных устройств.

Установлено, что во время тестирования функции системы работают в соответствии с требованиями. Ошибок, которые полностью препятствуют работе программы, обнаружено не было. Следовательно, данное программное обеспечение готово к использованию в полевых условиях.

Заключение

Таким образом, эффективная система управления и мониторинга производительности Интернета вещей должна соответствовать следующим требованиям:

- способна справиться с быстрым ростом трафика и объемов данных;
- может работать одновременно с протоколами IPv4 и IPv6;
- возможность управления всеми новыми устройствами, вне зависимости от стандарта связи или метрик производительности;
- видимость трафика в каждом сегменте с точностью до секунд;

- управление с одного экрана гибридной облачной средой, охватывающей все уровни мониторинга, включая физические, виртуальные, ключевые индикаторы производительности.

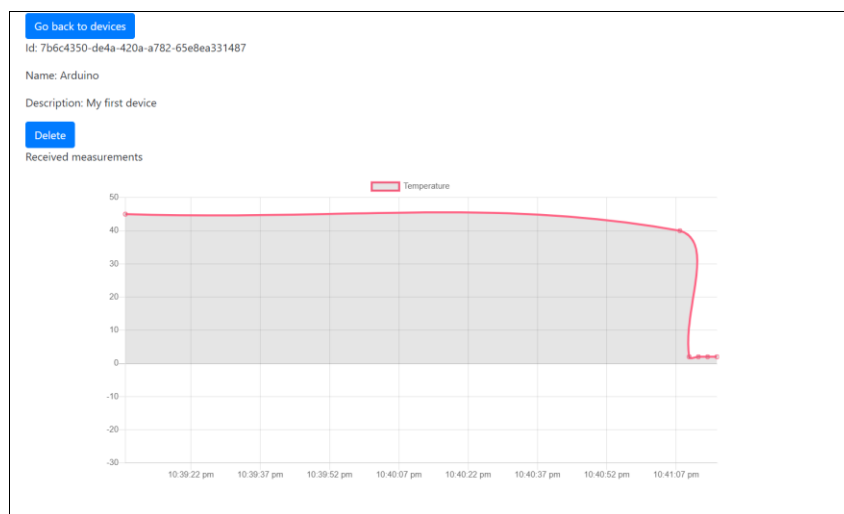


Рис.3. Страница описания устройства с графиком данных температуры, которые получены в реальном времени

Список литературы

1. Сарьян В. К., Саломатина Е. В., Лутохин А. С. Система индивидуализированного управления спасением пациентов e-helf в случае возникновения ЧС природного и техногенного происхождения. Труды международной конференции «Инжиниринг & Телекоммуникации - En&T 2015», Москва, МФТИ, ноябрь, 18-19, 2015.
2. Сарьян В.К., Сущенко Н.А., Лутохин А.С и др Прошлое, настоящее и будущее стандартизации Интернета вещей. Труды НИИР. №1 2014.
3. Bergenti, F., Caire, G., Gotta, D. Large-scale network and service management with WANTS. Industrial Agents: Emerging Applications of Software Agents in Industry. Elsevier, 2015. – 366 p.
4. Hu F., Security and Privacy in Internet of Things (IoT): Models, Algorithms, and Implementations, CRC Press, 2016. - 137 p.
5. Vermesan O, Friess P, Guillemin P., Sundmaeker H., Eisenhauer M., Moessner K., Le Gall F., and Cousin P., Internet of Things Strategic Research and Innovation Agenda, In River Publishers Series in Communications, Aalborg Denmark, 2013. - 158 p.

References

1. Sar'yan V. K., Salomatina E. V., Lutohin A. S. Sistema individualizirovannogo upravleniya spaseniem pacientov e-helf v sluchae vozniknoveniya CHS prirodnogo i tekhnogennogo proiskhozhdeniya. Trudy mezhdunarodnoj konferencii «Inzhiniring & Telekommunikacii - En&T 2015», Moskva, MFTI, noyabr', 18-19, 2015.
2. Sar'yan V.K., Sushchenko N.A., Lutohin A.S i dr Proshloe, nastoyashchee i budushchee standartizacii Interneta veshchej. Trudy NIIR. №1 2014.
3. Bergenti, F., Caire, G., Gotta, D. Large-scale network and service management with WANTS. Industrial Agents: Emerging Applications of Software Agents in Industry. Elsevier, 2015. – 366 p.
4. Hu F., Security and Privacy in Internet of Things (IoT): Models, Algorithms, and Implementations, CRC Press, 2016. - 137 p.
5. Vermesan O, Friess P, Guillemin P., Sundmaeker H., Eisenhauer M., Moessner K., Le Gall F., and Cousin P., Internet of Things Strategic Research and Innovation Agenda, In River Publishers Series in Communications, Aalborg Denmark, 2013. - 158 p.