

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 004.738

Кенда
Яна Васильевна

Веб-ориентированное программное средство управления электронными
устройствами

АВТОРЕФЕРАТ

на соискание академической степени
магистра информатики и вычислительной техники

по специальности 1-40 81 04 – Обработка больших объемов информации

Научный руководитель
Куликов С.С.
к.т.н., доцент

Минск 2020

КРАТКОЕ ВВЕДЕНИЕ

В современном обществе задаче автоматизации уделяется большое внимание. Все более актуальным становится создание программного обеспечения, которое не просто решает определенную задачу, а является универсальным. И таким образом подходит для решения большого спектра задач. Для перечисленных целей служит данное веб-ориентированное программное средство управления электронными устройствами с целью оптимизации работы бытовых устройств.

Программное средство предназначается для пользователей, которые хотят получить систему «умного дома», но при этом хотят иметь гибкую систему подключения устройств. Данное программное средство также может использоваться сотрудниками для производственных целей при добавлении соответствующих устройств к системе. Разберем более подробно вариант использования программного средства для системы «умного дома».

При отсутствии людей в доме или квартире зачастую возникает потребность в дистанционном управлении, проверке или контроле бытовой техники из другой точки города или даже из другой точки мира. Данные функции решаются реализованным программным средством управления электронными устройствами. Дистанционно пользователь, используя программное средство, может включать/выключать устройства и осуществлять контроль и мониторинг через просмотр записей работы подключенных устройств. Таким образом, решается вопрос контроля и дистанционного управления работой подключенных устройств, что помогает оптимизировать работу электронных устройств.

Также данное программное средство решает проблему планирования и организации работы устройств, так как с помощью данного приложения можно планировать повторяющиеся и единичные расписания для включения/выключения устройств в соответствии с пожеланиями и потребностями пользователя. Представьте, что свет выключается автоматически вечером в 12 ночи и включается утром в 8 утра в офисе, тем самым освобождая охранников офиса от обязанности выключать свет в кабинетах, или постоянно напоминать о необходимости выключения света работникам, или вывешивать знакомые таблички «Уходя, гасите свет».

Важным способом сбережения батареи устройств является зарядка до определенного процента без передержки устройства на подключенной зарядке, чтобы износ батареи был минимален. Для решения данной проблемы отлично подходит таймер обратного отсчета для включения/выключения устройств, что также представлен в функционале программного средства. Так, указав таймер, мобильный телефон перестает заряжаться через 3 часа, что позволяет избежать длительной зарядки и перегрева телефона, и как результат снижается износ батареи.

Еще одной особенностью программного средства является циклический таймер. Циклический таймер позволяет выполнять определенное действие на заданное количество времени в течение цикла времени. Например, с помощью

циклического таймера есть возможность осуществлять полив растений каждый день в 10 утра на протяжении 3 минут времени, что может являться оптимальным временем полива имеющегося растения.

Всем знакома ситуация, когда по приходу в квартиру или дом каждый раз необходимо выполнить один и тот же набор действий: включить свет, включить отопление для частного дома, к примеру, включить чайник, чтобы приготовить чай, или включить телевизор, если кто-то привык видеть телепередачи в фоне. По выходу из квартиры/дома также необходимо сделать ряд одинаковых действий: выключить свет во всей квартире/доме, проверить выключение утюга и другое. А теперь предположим, что это все делается автоматически по закрытию двери. Данную функцию также выполняет данное приложение, что значительно экономит время при выходе на работу, к примеру, что часто происходит в спешке у многих людей.

Таким образом, программное средство управления электронными устройствами позволяет оптимизировать работу бытовых устройств, сделать использование электронными устройствами безопасным, сэкономить время на управление и мониторинг электронно-бытовых устройств, а также за счет оптимизации работы устройств сэкономить на электроэнергии. Также данное программное средство является представителем дешевого сегмента, так как не требует больших затрат на установку и использование. Что позволяет сделать данное средство доступным для людей с низким уровнем достатка. И благодаря этому сделать возможным внедрение данного программного средства также людям с ограниченными возможностями, которые больше всего нуждаются в автоматизации, но не могут ввиду различных причин иметь средств на дорогие аналоги.

Программное средство управления электронными устройствами позволяет добавлять не только бытовые устройства, но и любые другие электронные устройства. Программное средство служит для «общения» подключенных устройств, а выполняемые программой функции зависят от подключенных к системе устройств. Описанная особенность делает программное средство универсальным и масштабируемым. Так, данное программное средство управления электронными устройствами подойдет не только для домашнего использования, а также может быть применено на производстве и в других отраслях.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Цель и задачи исследования

Целью диссертационной работы является разработка алгоритмов и программного обеспечения для решения задач управления электронными устройствами с целью оптимизации работы бытовых устройств.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Разработать концепцию системы по передаче данных между устройствами.
2. Провести сравнительный анализ возможных технологий для разработки каждого компонента системы.
3. Разработать архитектуру системы.
4. Реализовать программное обеспечение, реализующее поставленную на каждом этапе задачу.
5. Проанализировать надежность работы программного средства и подключаемых устройств.

Объектом исследования являются системы управления электронными устройствами.

Предметом исследования является программное обеспечение для подключения и управления электронными устройствами для дальнейшего взаимодействия устройств друг с другом.

Актуальность исследования заключается в том, что на сегодняшний день число устройств, подключенных к сети, превышает число всех жителей планеты и продолжает стремительно увеличиваться. Данный факт говорит о необходимости оптимизации работы этих устройств и, таким образом, об актуальности программного средства, которое являлось бы универсальной системой «общения» устройств, агрегирующей управление автоматизированными устройствами и доступной обычному пользователю без специализированных знаний.

Основной *гипотезой*, положенной в основу диссертационной работы, является возможность создания единого способа общения между устройствами с использованием платформы Arduino и MQTT-протокола для автоматизации выполнения бытовых задач.

Личный вклад соискателя

Результаты, приведенные в диссертации, получены соискателем лично. Вклад научного руководителя С.С. Куликова, заключается в формулировке целей и задач исследования.

Публикации

По теме диссертации опубликована 1 печатная работа в международном научном журнале.

Структура и объем диссертации

Диссертация состоит из введения, пяти глав, заключения, списка использованных источников, списка публикаций автора и приложений. В первой главе представлен анализ предметной области, выявлены основные существующие проблемы в рамках тематики исследования, показаны направления их решения. Вторая глава посвящена проектированию программного средства: а именно разработке архитектуры и функциональности ПС, проектированию модели подключаемых устройств. В третьей главе показан процесс реализации программного средства: рассмотрены используемые технологии, рассмотрено программирование микроконтроллеров, а также реализация серверной части программного средства. Четвертая глава посвящена тестированию программного средства, приведены тестовые сценарии, выполненные в рамках этапа тестирования. В пятой главе представлено руководство по установке и использованию программного средства, а также функционал по автоматизированному управлению устройств.

Общий объем работы составляет 79 страниц, из которых основного текста – 48 страниц, 31 рисунок на 24 страницах, 1 таблица на 8 страницах, список использованных источников из 13 наименований на 2 страницах и 2 приложения на 21 страницу.

ОСНОВНОЕ СОДЕРЖАНИЕ

Во **введении** определена область и указаны основные направления исследования, показана актуальность темы диссертационной работы, дана краткая характеристика исследуемой проблемы, обозначена практическая ценность работы.

В **первой главе** проведен анализ предметной области, а именно дана характеристика понятия Интернета вещей, системы «умного дома», а также проведен обзор аналогов программного средства.

Так, Интернет вещей – это концепция межмашинного взаимодействия, впервые сформулированная Кевином Эштоном в 1999 году. Суть концепции заключается в том, что все электронные устройства («вещи») объединяются в вычислительную сеть благодаря встроенным технологиям для передачи данных. При этом отдельные устройства могут как генерировать собственные данные, так и потреблять (преобразовывать) данные от других устройств с помощью соответствующих функций аппаратного и программного обеспечения.

На сегодня число IoT-гаджетов достигло 26,6 млрд. и продолжает стремительно увеличиваться.

В аналитическом исследовании компании IoT Analytics наиболее важными отмечены следующие применения Интернета вещей: экосистема «умного» города, программные системы для промышленных производственных систем, строительство, автомобильные информационные системы, энергопотребление, здравоохранение, «умные дома».

Разберем подробнее понятие «умный дом». Итак, под системой «умный дом» понимается объединение в единый согласованный комплекс всех подсистем и инженерного оборудования здания с возможностью централизованного и автоматизированного управления и контроля в целях повышения комфорта проживания и уровня безопасности. Таким образом, система «умный дом» выступает в качестве надстройки над остальными модулями – инженерными системами.

Преимущества «умного дома» – это комфорт, безопасность и экономия.

Системы и оборудование «умного дома» программируются специальным образом, так, чтобы рационализировать потребление всех ресурсов. До 35-40% снижаются расходы на электроэнергию, отопление, водопотребление, эксплуатацию дома или здания. При постоянном росте цен на ресурсы, средства, вложенные в автоматизацию, окупаются достаточно быстро: за 4-5 лет в Европе (данные взяты из аналитических материалов по Германии) и примерно за 10-15 лет в Беларуси при нынешней стоимости энергоносителей.

Также были рассмотрены аналоги программного средства, выявлены их преимущества и недостатки. Сравнительный анализ аналогов был учтен при реализации программного средства в рамках работы над диссертацией.

Вторая глава посвящена проектированию программного средства. Разработана программная архитектура, проведен анализ необходимой аппаратной конфигурации, описана планируемая функциональность и спроектированы модели подключаемых к системе устройств.

Для описания аппаратной конфигурации, на которой будут работать части конечного программного средства, была использована диаграмма развертывания (рисунок 1).

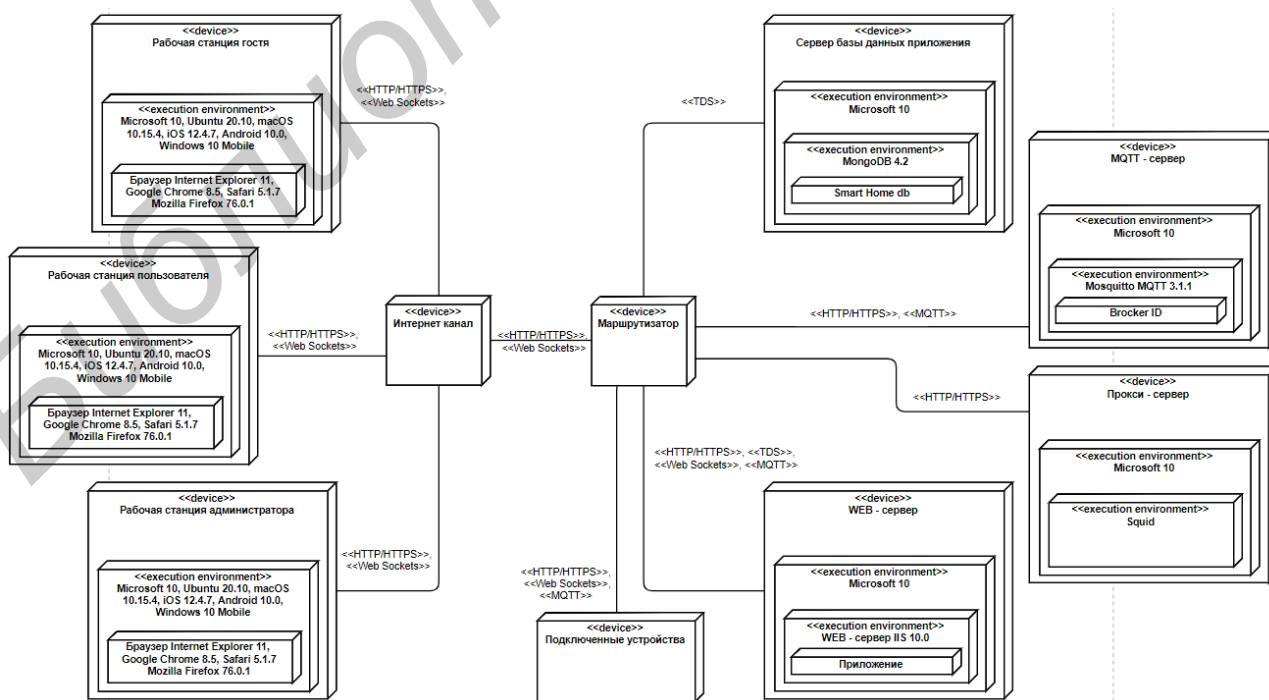


Рисунок 1 – Диаграмма развертывания

Для представления функциональной модели была выбрана диаграмма вариантов использования UML, которая отражает отношения между актерами и прецедентами и позволяет описать систему на концептуальном уровне. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Диаграмма вариантов использования разрабатываемого программного средства представлена на рисунке 2.

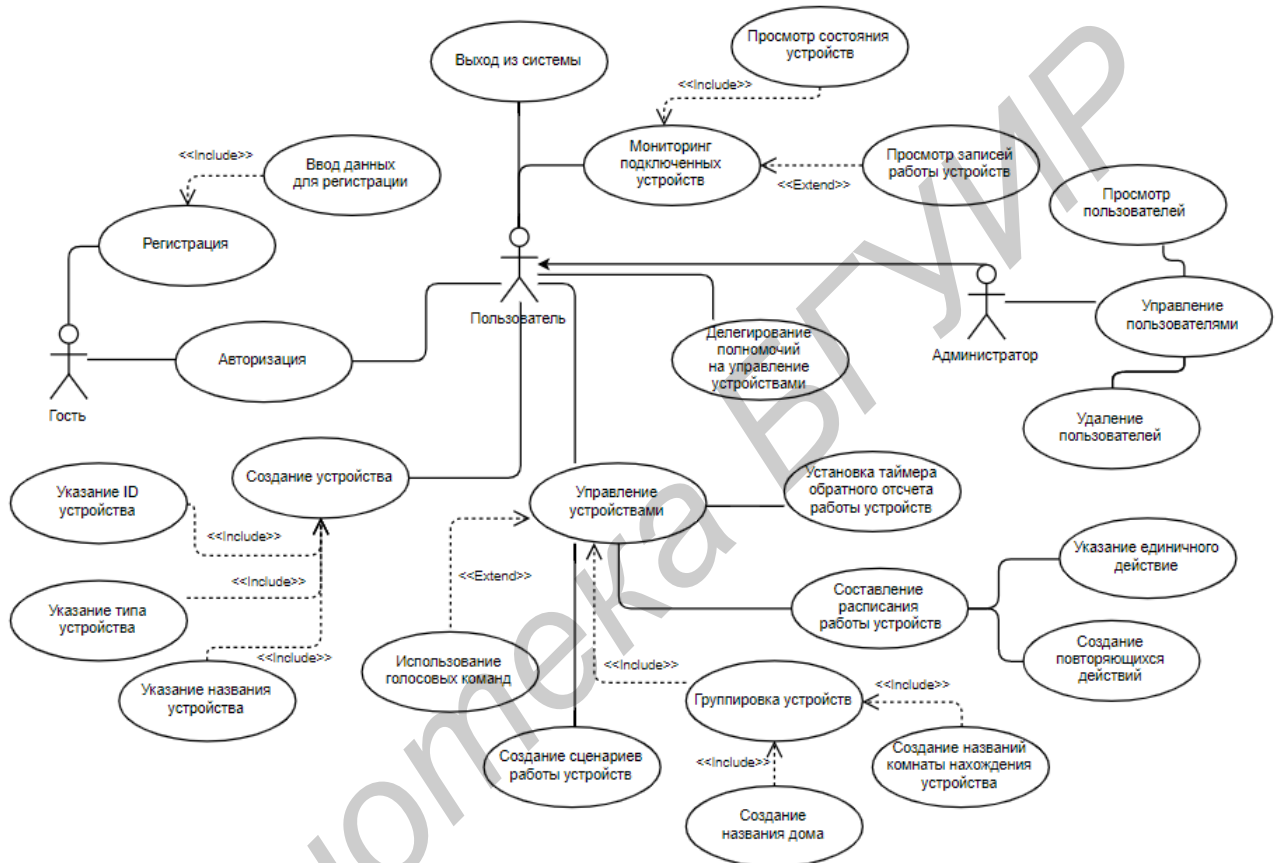


Рисунок 2 – Диаграмма вариантов использования

Также было проведено проектирование модели подключаемых устройств. Так, устройства, подключаемые к системе, можно разделить по типу взаимодействия с ними и по типу используемых вспомогательных устройств на следующие категории:

- а) дискретное включение/выключение устройства с использованием реле. Применяется, к примеру, для включения и выключения лампы;
- б) плавное регулирование работы устройства с использованием диметра. Может быть применено для регулирования подачи света от лампы;
- в) взаимодействие с устройством с помощью инфракрасного сигнала. Применяется для подключения к телевизору или кондиционеру;
- г) приведение в действие устройства с использованием сервопривода с мотором. Может быть применено для поднятия и опускания штор;

д) сбор данных с устройства с помощью датчиков (датчиков задымления, затопления, утечки газа, освещенности и других).

Для данного программного средства будут применены дискретное включение/выключение устройства, взаимодействием с устройством с помощью инфракрасного сигнала, а также сбор данных с устройства с помощью датчиков.

В третьей главе предложены методы и алгоритмы реализации программного средства, проведен обзор применяемых технологий, описано программирование микроконтроллеров.

Для реализации программного средства были применены следующие технологии: React Redux, Node JS, Express, Redux Saga. База данных – MongoDB. Для передачи данных использовался MQTT протокол, Web Sockets.

Аппаратная часть программного средства включает в себя использование Arduino и Raspberry Pi.

Для описания работы программного средства следует дать характеристику работы MQTT протокола. Протокол MQTT (Message Queuing Telemetry Transport) – это протокол для передачи последовательности сообщений с телеметрическими данными, то есть информации от датчиков температуры, влажности, освещенности и др. В программном средстве данный протокол используется для «общения» устройств друг с другом и с брокером.

Обмен сообщениями, согласно данному протоколу, происходит по принципу «издатель-подписчик». Отличительной особенностью данного принципа от клиент-серверного подхода является то, что клиенты, посылающие сообщения (издатели, Publisher), и клиенты, принимающие сообщения (подписчики, Subscriber), как правило, разделены. Издатель и подписчик не передают сообщения напрямую, не устанавливают прямой контакт, могут не знать о существовании друг друга. Координирует и управляет передачей сообщений от издателя к подписчику и от подписчика к издателю брокер (Broker). Распараллеливание операций на брокере является второй важной особенностью принципа взаимодействия «издатель-подписчик».

MQTT-клиент – это устройство, оснащенное микроконтроллером, поддерживающим стек TCP/IP.

Брокер является основным элементом системы «издатель-подписчик». Он отвечает за прием и фильтрацию сообщений, принятие решения о том, кому интересны сообщения, и, в конечном итоге, за пересылку сообщений клиентам-подписчикам.

Таким образом, множество подписчиков может быть подписано на разнообразные темы и в зависимости от подписок получать необходимую информацию, не общаясь с издателем напрямую. Так осуществляется передача информации по принципу «издатель-подписчик» (рисунок 3).

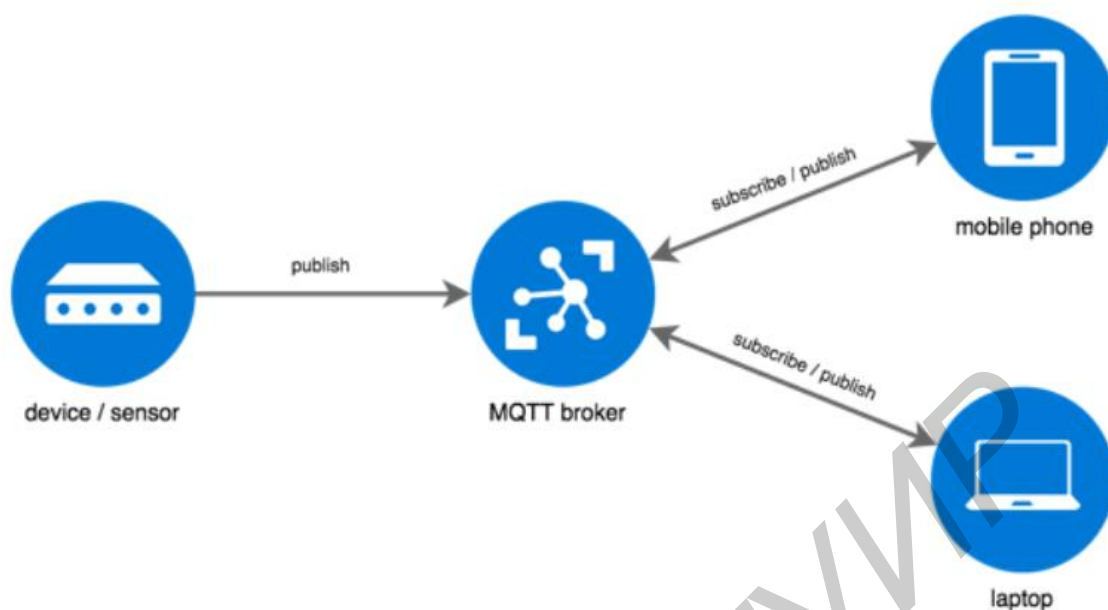


Рисунок 3 – Протокол MQTT

Диаграмма последовательности отображает схему работы MQTT протокола для данного программного средства (рисунок 4).

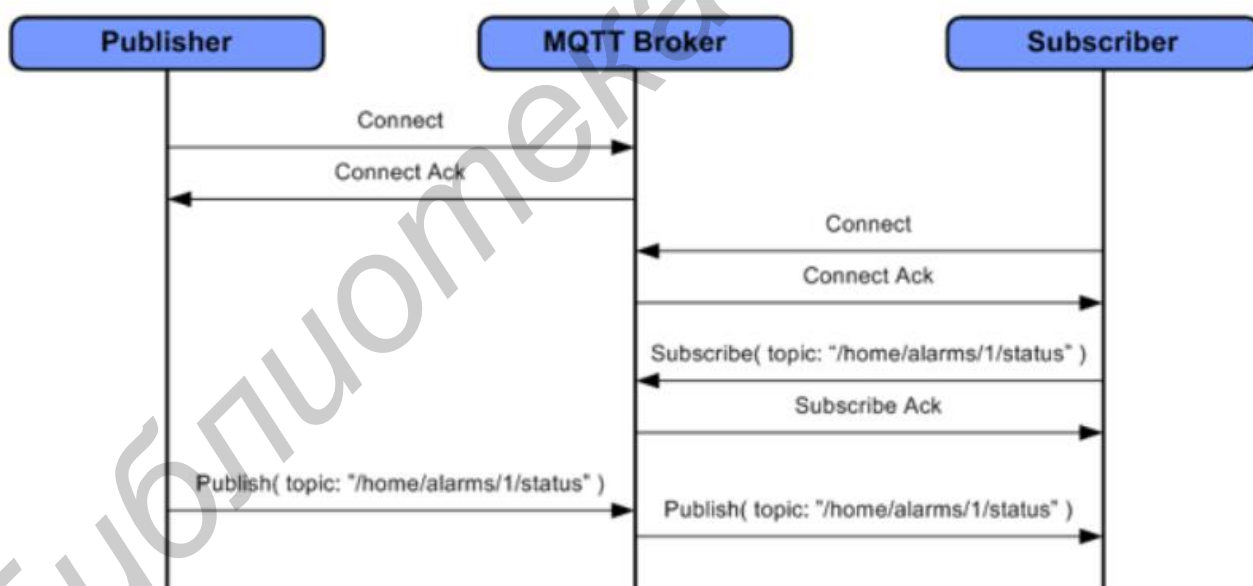


Рисунок 4 – Диаграмма последовательности работы MQTT протокола для программного средства

Разобравшись с MQTT-протоколом, необходимо дать определение понятию Arduino. Arduino – это открытая программируемая аппаратная платформа для работы с различными физическими объектами и представляет собой простую плату с микроконтроллером, а также специальную среду разработки для написания программного обеспечения микроконтроллера.

Программная часть состоит из бесплатной программной оболочки (IDE) для написания программ, их компиляции и программирования аппарату-

ры. Аппаратная часть представляет собой набор смонтированных печатных плат.

Arduino может использоваться для разработки интерактивных систем, управляемых различными датчиками и переключателями. Такие системы, в свою очередь, могут управлять работой различных индикаторов, двигателей и других устройств. Проекты Arduino могут быть как самостоятельными, так и взаимодействовать с программным обеспечением, работающем на персональном компьютере (например, приложениями Flash, Processing, MaxMSP). Любую плату Arduino можно собрать вручную или же купить готовое устройство; среда разработки для программирования такой платы имеет открытый исходный код и полностью бесплатна.

И также было определено понятие ESP8266. ESP8266 – это микроконтроллер, разработанный в 2014 году и выпускающийся компанией Espressif Systems – китайской компанией из Шанхая. Он представляет собой сетевое решение с Wi-Fi-трансивером на борту плюс возможность выполнения записываемых в его память приложений.

Применение модуля ESP8266:

- управляемые розетки;
- домашняя автоматизация;
- промышленное беспроводное управление;
- IP-камеры;
- сети датчиков;
- портативная электроника;
- решения в отрасли безопасности.

Для аппаратной части реализованного программного средства управления электронными устройствами использовался одноплатный компьютер Raspberry Pi.

Raspberry Pi – одноплатный компьютер размером с банковскую карту. Разрабатывается британской компанией Raspberry Pi Foundation. Изначально проект создавался как образовательный, Raspberry Pi отлично подходит для изучения основ электроники. Однако сегодня его назначение вышло за только образовательное. На сегодняшний день данный одноплатный компьютер используется также в производственных целях. Используется для репликации данных при отсутствии интернета. Локальный сервер и сервер в облаке синхронизируются.

Также данная глава включает в себя раздел с описанием программирования микроконтроллеров. В данном разделе было продемонстрировано подключение устройства к веб-ориентированному программному средству управления электронными устройствами. Описанный принцип применяется и к другим подключаемым устройствам системы с добавлением MQTT протокола для передачи запроса устройству. Отличительной особенностью данного метода является то, что на устройство заранее будет загружена микропрограмма. А конечному пользователю будет необходимо лишь подключить устройство через последовательный порт к компьютеру, выполнить подключение к точке доступа устройства и указать данные устройства для подключения к сети.

Код микропрограммы на языке C++ смотрите в приложении А.

Данная глава также включает описание реализации серверной части программного средства. Продемонстрирован код по получению информации от подключенных устройств к веб-ориентированному программному средству управления электронными устройствами, представлен код по настройке работы с MQTT-протоколом с использованием Mosco и продемонстрирован код модели и контроллера для подключаемых устройств. Полный код серверной части программного средства смотрите в приложении В.

Четвертая глава посвящена тестированию программного средства. Были применены модульное и интеграционное тестирования, разработаны и пройдены тестовые сценарии, использованные при разработке программного средства, а также применены мануальные тесты, проведенные над реализованной системой.

Для автоматизации тестирования использовался фреймворк Mocha с использованием библиотеки Chai для выполнения проверок, предназначенных для запуска модульных и интеграционных тестов, а также агрегации результатов выполнения.

Таким образом, тест-кейсы покрыли авторизацию, регистрацию и некоторый функционал по управлению электронными устройствами. Успешность прохождения тестов показывает корректность работы программы с реальными данными и соответствие функциональным требованиям.

Пятая глава содержит руководство по установке и использованию веб-ориентированного программного средства управления электронными устройствами, описание требований к системе, иллюстрации веб-интерфейса программного средства, а также руководство по автоматизированному управлению электронными устройствами.

Разработанное программное средство управления электронными устройствами нетребовательно к аппаратной части. И может быть запущено через браузер на компьютере, телефоне или планшете.

Таким образом, могут быть использованы следующие ОС:

- Microsoft 10;
- Ubuntu 20.10;
- macOS 10.15.4;
- iOS 12.4.7;
- Android 10.0;
- Windows 10 Mobile и другие.

И использованы следующие браузеры для запуска веб-ориентированного программного средства управления электронными устройствами:

- Internet Explorer 11;
- Google Chrome 8.5;
- Safari 5.1.7;
- Mozilla Firefox 76.0.1 и другие.

Работа с сайтом начинается с главной страницы, где можно авторизоваться для дальнейшей работы. Если нет логина и пароля, необходимо выполнить регистрацию. Для этого нажимается кнопка «Register», и далее отобразится стра-

ница регистрации на сайте. После заполнения данных и отправки данных формы, пользователь будет зарегистрирован в веб-ориентированном программном средстве управления электронными устройствами.

В системе существует 3 роли:

- гость;
- пользователь;
- администратор.

И в зависимости от роли открываются те или иные полномочия в системе. Для доступа к приложению и для осуществления работы с ним гостю необходимо зарегистрироваться. Став пользователем, для него открываются полноценные возможности для управления устройствами через систему. Администратор в свою очередь обладает правами на управление пользователями, а именно администратор может просматривать пользователей, зарегистрированных в системе, а также имеет право на их удаление. Таким образом, вкладка по управлению пользователями доступна исключительно для роли администратора программного средства.

К основным функциям данного программного средства относятся добавление устройств в систему и их группировка по домам и комнатам. Для того, чтобы добавить устройство необходимо указать название дома «House» и название комнаты «Room». Для этого пользователь переходит во вкладку «My Houses». На экране отображаются существующие названия домов и опция создания нового названия дома для добавления его в систему. Для этого необходимо выбрать опцию «New house...».

Далее открывается форма для заполнения информации о доме: названии дома и идентификаторе брокера. Для сохранения указанной информации нажимается кнопка «Add».

Для задания названия комнат дома и устройств нажимается кнопка на изменение напротив названия дома.

Открывается страница для добавления комнат дома. Чтобы добавить комнату необходимо написать ее название и нажать кнопку «Add». Заданная комната отобразится на странице просмотра названий комнат дома.

Следующим шагом является добавление подключенных к сети интернет устройств. Для этого выбирается список выпадающих значений напротив комнаты, для которой необходимо добавить устройство, и далее выбирается опция добавления устройства – «New device...».

Далее открывается форма для заполнения информации об устройстве: названии устройства, идентификаторе устройства и типе устройства. Тип устройства выбирается из списка преднастроенных значений:

- «Light Switcher»;
- «IP Cam»;
- «Climate Sensor».

Данный список будет расширяться с течением времени по мере развития данного программного средства. Для сохранения указанной информации нажимается кнопка «Add».

Созданные устройства отображаются на странице по управлению устройствами дома. Рассмотрим управление устройствами на примере лампы. Для включения света лампы нажимается кнопка активации напротив устройства – кнопка становится выделена соответствующим зеленым цветом. При необходимости выключения лампы нажимается кнопка деактивации напротив устройства – подсветка кнопки переходит в характерный серый цвет. Соответственно устройство, в данном случае лампа, включается или выключается согласно заданным действиям.

После того, как устройства были заданы в программном средстве, необходимо выполнить подключение устройств к системе.

В данной главе также представлена, реализованная в рамках данного проекта, функция автоматизированного управление светом (включение/выключение) в зависимости от присутствия людей в помещении. Соответственно при входе человека в помещении выполняется включение света, при выходе – выключение. Программа также учитывает количество людей в комнате. То есть выполняется суммирование количества людей, вошедших в комнату, а также уменьшается количество людей при выходе из комнаты.

Для реализации данного функционала использовались Arduino Uno, Raspberry Pi, 2 лазера, 2 фотоприемника, реле для включения и выключения лампы в зависимости от подсчетов системы о нахождении там людей, а также модуль SD карты для логирования работы устройств.

В рамках данной диссертационной работы была добавлена функция голосового управления устройствами. Для данной цели использовалось настраиваемое решение IFTTT и Google Ассистент (Google Assistant). Использование данных решений стало возможно благодаря предоставлению реализованным программным средством API для сторонних сервисов.

Таким образом, были заданы условия по включению и выключению света при распознавании Google Ассистентом слов «light on» и «light off» соответственно в сервисе IFTTT. Было также установлено реле к лампе для выполнения действий над устройством. Заданы следующие значения: при передаче значения 1 лампа включается, при передаче значения 0 – выключается. Так, были сформированы URL для отправки запроса к серверу приложения, а от сервера к лампе, подключенной к сети интернет.

Так же была реализована функция установки температуры на заданном устройстве (кондиционере) с использованием голосовой команды. Так, при произношении голосовой команды «Set temperature to # degrees», где вместо # необходимо назвать температуру в градусах по Цельсию, происходит установка программы кондиционера на заданную температуру. На кондиционере для приема заданных команд установлен инфракрасный приемник, а также предварительно считаны и загружены на сервер коды команд для конкретной модели кондиционера. Таким образом, были сформированы URL для отправки запроса к серверу приложения, а от сервера сформированы и отправлены необходимые коды команды к инфракрасному передатчику, расположенному напротив кондиционера, а от него уже к инфракрасному приемнику кондиционера для выполнения команд.

В дальнейшем планируется внедрение своей модели по распознаванию речи и произносимых пользователем команд.

ЗАКЛЮЧЕНИЕ

Основные научные результаты диссертации

1. Исследованы методы решения задачи подключения и управления электронными устройствами для дальнейшего взаимодействия устройств друг с другом. Был изучен и внедрен протокол MQTT для передачи данных между электронными устройствами, а также MQTT over Websockets и MERN стек. Реализована универсальная система «общения» устройств, доступная обычному человеку без специализированных знаний. Проведено тестирование разработанного программного средства.

2. Исследованы перспективы и причины для применения концепции и технологий в решении вопроса агрегации управления автоматизированными устройствами.

3. Разработана и обоснована архитектура программного средства, призванного решить проблему оптимизации работы устройств. Данная система реализована в качестве веб-приложения.

Рекомендации по практическому использованию результатов

1. Полученные результаты формируют достаточную теоретическую и практическую базу для разработки программного обеспечения с использованием концепции Интернета вещей на основе MQTT-протокола. Кроме того, они могут быть использованы для дальнейшего исследования и развития представленной системы.

2. Разработанная архитектура приложения и подходы могут применяться не только в поставленной задаче оптимизации работы бытовых устройств, но и в любой другой, направленной на оптимизацию работы устройств и их взаимодействия. На основе разработанной архитектуры возможно создание системы любой сложности и масштаба.

3. Результаты работы могут использоваться при подготовке специалистов в области программного обеспечения и к разработке решений для Интернета вещей.

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ

1-А. Кенда Я.В. Программное средство управления электронными устройствами // Студенческий вестник: научный журнал. – № 19(117). – М., Изд. «Интернаука», 2020.