

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ
ГУМАРБЕКА ДАУКЕЕВА»

Институт космической инженерии и телекоммуникаций
Кафедра электроники и робототехники

«ДОПУЩЕН К ЗАЩИТЕ»

Зав.кафедрой доц., Чигамбаев Т.О.

_____ «__» _____ 2020г.
(подпись)

ДИПЛОМНАЯ РАБОТА

На тему: «Разработка системы контроля доступа»

Специальность «5В071600 – Приборостроение»

Выполнила Опарина И.А. Группа ПС-16-4

Научный руководитель доц., Байкенов Б.С.

_____ «__» _____ 2020г.
(подпись)

Консультанты:


по экономической части: ст.преп., Боканова Г.Ш.

_____ «__» _____ 2020г.
(подпись)

по безопасности жизнедеятельности: к.т.н., доц., Приходько Н.Г.

_____ «__» _____ 2020г.
(подпись)

Нормоконтролер: ст. преп., Фазылова А.Р.

 _____ « 20 » _____ 05 2020 г.
(подпись)

Рецензент: д.т.н., проф. Байбатшаев М.Ш.

_____ «__» _____ 2020г.
(подпись)

Алматы 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ
ГУМАРБЕКА ДАУКЕЕВА»

Институт космической инженерии и телекоммуникаций
Кафедра электроники и робототехники
Специальность «5В071600 – Приборостроение»

ЗАДАНИЕ

на выполнение дипломного проекта (работы)

Студентке Опариной И.А.

Тема работы (проекта) «Разработка системы контроля доступа»

Утверждена приказом по университету № ____ от «____» _____ 20__г.

Срок сдачи законченной работы (проекта) «____» _____ 2020г.

Исходные данные к работе (требуемые параметры результатов исследования (проектирования) и исходные данные объекта):

1. Микрокомпьютер Raspberry Pi 3 B+;
2. RFID-считыватель RC522;
3. Светодиод;
4. Макетная плата;
5. Соединительные провода
6. Элементы для сборки.

Перечень вопросов, подлежащих разработке в дипломной работе, или краткое содержание дипломной работы:

1. Анализ системы контроля доступа и посещаемости (описание существующих систем);
2. Разработка системы учета посещаемости (разбор компонентов и построение схем);
3. Программное средство (алгоритмы и листинг программы);
4. Безопасность жизнедеятельности (расчет освещения и шума в помещении);
5. Экономическая часть (разработка бизнес-плана).

Перечень графического материала (с точным указанием обязательных чертежей): в данной дипломной работе содержатся рисунки и таблицы.

Основная рекомендуемая литература:

1. Петин В.А. Arduino и Raspberry Pi в проектах Internet of Things, 2-е изд. СПб.: БХВ-Петербург, 2019. - 432 с.
2. Макаров С. Л. Arduino Uno и Raspberry Pi 3: от схемотехники к интернету вещей. – Москва.: ДМК Пресс, 2018. - 204 с.
3. Шварц Б., Зайцев П., Ткаченко В. MySQL по максимуму. Санкт-Петербург, 2018. - 864 с.

Консультации по работе (проекту) с указанием относящихся к ним разделов работы (проекта)

Раздел	Консультант	Сроки	Подпись
Анализ системы контроля доступа и посещаемости	Байкенов Б.С.	30.01	
Разработка системы учета посещаемости	Байкенов Б.С.	23.02	
Программное средство	Байкенов Б.С.	12.04	
Безопасность жизнедеятельности	Приходько Н.Г.	22.05	
Бизнес-план	Боканова Г.Ш.	22.05	

График
подготовки дипломного проекта (работы)

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
1 Анализ системы контроля доступа и посещаемости	16.01	
1.1 Обзор системы контроля доступа	30.01	
1.2 Обзор и выбор актуальной платформы	30.01	
1.3 Подготовка к созданию системы контроля доступа	30.01	
1.4 Настройка SPI	30.01	
1.5 Установка и настройка NGINX и PHP	30.01	
1.6 Подключение и настройка Git и SSH	30.01	
2 Разработка системы учета посещаемости	16.02	
2.1 Постановка задачи	16.02	
2.2 Общее соединение и вид устройства	23.02	
2.3 Создание схемы считывателя RFID RC522	23.02	
3 Программное средство	02.03	
3.1 Тестирование RFID RC522	02.03	
3.2 Подготовка базы данных системы посещаемости RFID	02.03	
3.3 Запись пользователя в системе посещаемости	10.03	
3.4 Запись посещаемости	10.03	
3.5 Проверка базы данных	15.03	
3.6 Создание веб-интерфейса	15.03	
4 Безопасность жизнедеятельности	25.04	
5 Бизнес-план	25.04	
Заключение	26.04	
Список литературы	26.04	

Дата выдачи задания « ____ » _____ 2020г.

Заведующий кафедрой _____ Чигамбаев Т.О.
(подпись)

Научный руководитель
работы (проекта) _____ Байкенов Б.С.
(подпись)

Задание принял к
исполнению студент _____ Опарина И.А.
(подпись)

Аннотация

В данной дипломной работе рассмотрены технологические аспекты системы контроля доступа и учета посещаемости. Описана работа и применение микрокомпьютера Raspberry Pi 3, RFID-считывателя и светодиода, которые относятся к данной дипломной работе. Система контроля доступа была разработана с использованием данных устройств.

В работе рассмотрены вопросы безопасности жизнедеятельности, охраны труда, составлен бизнес-план по внедрению данной системы.

Андатпа

Бұл дипломдық жұмыста кіруді бақылау және келуді есепке алу жүйесінің технологиялық аспектілері қарастырылған. Осы дипломдық жұмыста Raspberry Pi 3 микрокомпьютері, RFID-оқитын құрылғы және жарық диодының жұмысы мен қолданылуы сипатталған. Рұқсатты бақылау жүйесі осы құрылғыларды пайдалану арқылы әзірленді.

Жұмыста тіршілік қауіпсіздігі, еңбекті қорғау мәселелері қарастырылды, осы жүйені енгізу бойынша бизнес-жоспар жасалды.

Summary

The technological aspects of the access control and attendance accounting system have been considered in this work. The operation and application of the Raspberry Pi 3 microcomputer, RFID reader and LED, which are related to this graduate work, were described. The access control system was developed using these devices.

The project considers the issues of life safety, labor protection, and a business plan for the implementation of this system.

Содержание

Введение	8
.....	
1 Анализ системы контроля доступа и посещаемости	10
.....	
1.1 Обзор системы контроля доступа	10
.....	
1.2 Обзор и выбор актуальной платформы	11
.....	
1.3 Подготовка к созданию системы контроля доступа	12
1.4 Настройка SPI	16
.....	
1.5 Установка и настройка NGINX и PHP	18
.....	
1.6 Подключение и настройка Git и SSH	26
.....	
2 Разработка системы учета посещаемости	28
.....	
2.1 Постановка задачи	28
2.2 Общее соединение и вид устройства	30
.....	
2.3 Создание схемы считывателя RFID RC522	31
.....	
3 Программное средство	33
.....	
3.1 Тестирование RFID RC522	33
.....	
3.2 Подготовка базы данных системы посещаемости RFID	35
3.3 Запись пользователя в системе посещаемости	39
.....	
3.4 Запись посещаемости	42
3.5 Проверка базы данных	44
.....	
3.6 Создание веб-интерфейса	45
4 Безопасность жизнедеятельности	49
.....	
4.1 Анализ опасных и вредных факторов	49
.....	
4.1.1 Опасные и вредные факторы, воздействующие на персонал в процесс проектирования и эксплуатации объекта	49
.....	
4.1.2 Микроклимат. Влияние компьютера на рабочего	50
.....	

4.2 Расчетная часть	54
4.2.1 Расчет искусственного освещения	54
4.2.2 Расчет уровня шума, производимого персональным компьютером	57
5 Бизнес-план	61
5.1 Резюме	61
5.2 Описание продукции	61
5.3 Анализ рынка сбыта	61
5.4 Инвестиционный план	62
5.4.1 Расчет капитальных вложений	62
5.4.2 Расчет трудоемкости разработки программного обеспечения ...	63
5.4.3 Расчет затрат на разработку программного обеспечения	66
5.4.4 Расчет эксплуатационных расходов	69
5.4.5 Расчет доходов, прибыли чистой прибыли и окупаемости	69
Заключение	72
Список литературы	73

Введение

Разработка системы контроля доступа является актуальным решением для отрасли приборостроения в связи с ростом потребительских нужд, увеличением обширности автоматизации различных ветвей производства и нарастанием филиалов крупных компаний. Данный проект позволяет упростить сбор информации о работниках компаний путем автоматического считывания данных; улучшить работоспособность на предприятиях, снизить трудозатраты, а также увеличить скорость внедрения актуальных и действенных решений в случае возникновения соответствующих неблагоприятных условий, мешающих качественному течению рабочего процесса.

Система контроля доступа направлена на отслеживание и предоставление доступа при наличии специализированной id-карты, также обладает возможностью ведения учета посещаемости пользователей, имеющих доступ к данной системе, что, в свою очередь, позволяет руководству в любой необходимый момент времени получить всю необходимую информацию о конкретном пользователе: номер id-карты, имя,

на которое зарегистрирована карта; точная дата посещения с указанием времени до секунд.

Данный механизм обеспечивает возможность упрощения работы по сбору информации о посещениях работников, мониторингу за персоналом, выявлению отсутствия должной и требуемой дисциплины среди подчиненных. Следовательно, это позволяет принимать на работу добросовестных, компетентных и организованных специалистов, в том числе избежать финансовой нестабильности и нерационального использования денежных средств, обрести инстинктивно созданную сплоченность и организованность коллектива.

Существенную роль в реализации проекта представляет создание надежного веб-сервера, имеющего бесперебойное питание и доступ к интернету, с помощью чего интересующую информацию о посещаемости и доступе можно получить за пару кликов мыши. За данными можно обратиться и с любых других устройств, имея свободный доступ к интернету и адрес веб-сервера, что упрощает отношения пользователя с системой и создает условия её рационального бесконтактного использования. В основе проекта лежит создание сервера с устойчивой базой данных путем применения программирования в среде реляционной системы управления базами данных MySQL, и её совместная работа с RFID-считывателем, являющимся передатчиком данных для хранения их в соответствующей информационной системе.

Возможностью для упрощенного создания данной системы является факт существования микрокомпьютера Raspberry Pi, который, в отличие от прочих ныне действующих платформ, способен автономно работать полный рабочий день, при условии наличия специальных охлаждающих устройств и комплектующих (таких как медные/алюминиевые радиаторы); и при всем этом не нуждается в особых вложениях, обеспечивает стабильную и надежную работу.

Основой для проекта является модель Raspberry Pi 3 B+, отличающаяся от предшественников более расширенными возможностями в создании различных проектов и мощностью процессора, что позволяет создавать усложненные системы, которые микрокомпьютер в состоянии реализовать без снижения скорости их выполнения и качества.

1 Анализ системы контроля доступа и посещаемости

1.1 Обзор системы контроля доступа

Создание систем, управляющих объектами производства, развивается с каждым годом, поэтому целесообразно усовершенствование и внедрение обновленных «умных» механизмов, которые облегчат работу компаний в борьбе за безопасность, представляя собой совокупность программно-аппаратных технических средств. Основополагающей задачей системы контроля доступа является фильтрация пользователей по признаку разрешения доступа.

Проект позволяет выполнять несколько функций, таких как:

- а) ограничение доступа;
- б) идентификация;
- в) учет времени;
- г) внесение информации в базу данных;
- д) мониторинг данных с любого устройства.

Главным элементом работы выступает одноплатный микрокомпьютер Raspberry Pi, автономно работающий длительное время, тем самым гарантируя постоянство и отсутствие погрешностей. Raspberry Pi позволяет выполнять функции, доступные мощным системам, обладая большими графическими возможностями; подходит для использования в качестве платформы для разработки приложений, а также усложненных систем, затрагивающих измерения и робототехнику, с применением исполнительных механизмов и датчиков. При разработке нескольких систем разного характера отсутствует потребность в дополнительном микрокомпьютере – достаточно использовать для иной разработки карту памяти SD, тем самым исключая необходимость в избавлении от предыдущего исследования. Все модели поддерживают стабильность и возможность управления, но самым значительным обновлением Raspberry Pi 3 является 64-битный процессор с номинальной частотой 1.4 ГГц, что подразумевает быстроедействие в 10 раз превышающее скорость работы первой версии данной серии микрокомпьютеров, в среднем на 50-60% быстрее процессора Raspberry Pi 2.

Приоритетными критериями любой системы автоматики значатся: составление основного алгоритма и применение библиотек; учет дискретизации и скорости опроса аналоговых сигналов; учет скорости цикла обработки операции; наличие сетевой связи.

В данный проект при желании можно добавить жидкокристаллический дисплей для улучшения визуализации системы, отслеживания обращений к пользователям и контроля исправности механизмов. Одним из ключевых элементов является идентификатор – карточка, брелок, метка – базовая часть сборки системы контроля доступа, так как хранит код, служащий для определения прав («идентификации») владельца. В этой работе используется бесконтактная карта – RFID-метка. Тип используемого идентификатора

определяет в значительной степени надежность системы (устойчивость к взлому). Тип считывателя (устройства, которое получает код идентификатора и передает его в контроллер) определяется типом идентификатора. RFID-считыватель RC522, используемый в проекте, имеет в комплекте соответствующий идентификатор.

Контроллер не может работать бесперебойно неделями, для поддержки системы можно дополнить ее вентиляторами, либо иными охлаждающими устройствами, дабы уменьшить температуру нагрева процессора и отслеживать максимальный уровень производительности, тем самым добиваясь стабильной работы объектов и созданного механизма в целом. При необходимости обеспечения работы контроллера при авариях электросети, блок контроллера оснащается собственным аккумулятором, либо внешним блоком резервного питания.

1.2 Обзор и выбор актуальной платформы

При подобном уровне развития технологий, которого мир достиг на сегодняшний день, сфера электроники предоставляет широкий набор одноплатных компьютеров, имеющих различные функции, преимущества и недостатки. Создание системы контроля доступа и учета посещаемости подразумевает разработку устойчивой системы, поэтому одной из важных частей проекта стал одноплатный компьютер Raspberry Pi 3 B+, схожий по размеру с обычной банковской картой (рисунок 1.1). Raspberry позволяет выполнять и реализовывать широкий спектр функций, которые доступны мощным настольным системам; также отличается от конкурентных систем стабильностью работы, наличием модуля Bluetooth и модулей беспроводного интернета.

Операционная система прописывается на microSD и вставляется в специальный слот. Основной операционной системой на Raspberry является Linux, однако микрокомпьютер позволяет установить одну из множества операционных систем, таких как: Raspbian (производная Debian, но подстроена под Raspberry), Raspberry PI Desktop, Windows 10 IOT, OSMC, UBUNTU Mate, Snappy UBUNTU Core, PINET, LIBREELEC, Weather station, RISC OS.

Значительным отличием и обновлением в сравнении с предыдущими моделями Raspberry является 64-битный процессор ARM Cortex-A53, обладающий номинальной частотой в 1.4 ГГц, в том числе имеющий 4 ядра; оперативная память RAM 1024 МБ интегрирована в CPU (одним из удобных плюсов выступает установка памяти пользователем и наличие нескольких карт памяти, что в значительной степени упрощает работу пользователя).

Технические характеристики также включают в себя:

- а) графический процессор VideoCore IV;
- б) встроенный двухдиапазонный Wi-Fi (2.4 ГГц/ 5 ГГц) стандарта IEEE 802.11 b/g/n/ac (чип Cypress CYW43455);
- в) встроенный Bluetooth 4.2 LE; четыре порта на USB 2.0;

- г) полноразмерный HDMI-разъем;
- д) композитный 3.5 мм jack для вывода звука/видео;
- е) 40 пинов входа-выхода GPIO;
- ж) интерфейс для подключения камеры CSI и дисплея DSI.

Электропитание у данной модели от microUSB на 5 В, от 700 мА, что говорит о низком потреблении энергии. Отсутствие периферии при приобретении устройства решается путем использования элементов аппаратуры персонального компьютера, так как они вполне подходят для использования совместно с данным одноплатным компьютером.

Среди доступных интерфейсов, которые имеются у иных одноплатных компьютеров можно выделить: USB, HDMI, Gigabit Ethernet, UART, GPIO, SPI, DSI, DSIM, JTAG, Wi-Fi, I2C. Вес Raspberry Pi 3 B+ равен около 45 г, разработчиком CPU выступает Broadcom [1].

BCM2837B0 в целях улучшения производительности и экономии энергии был усовершенствован контролем питания. Имея рабочую температуру до 70°C работа ядер фиксируется на максимальной частоте 1.4 ГГц для достижения максимальной производительности, при температуре свыше 70°C частота ядер снижается до 1.2 ГГц. Данное внедрение не позволяет чипу достигать критических температур в ходе работы, сохраняя при всем этом оптимальный показатель производительности.

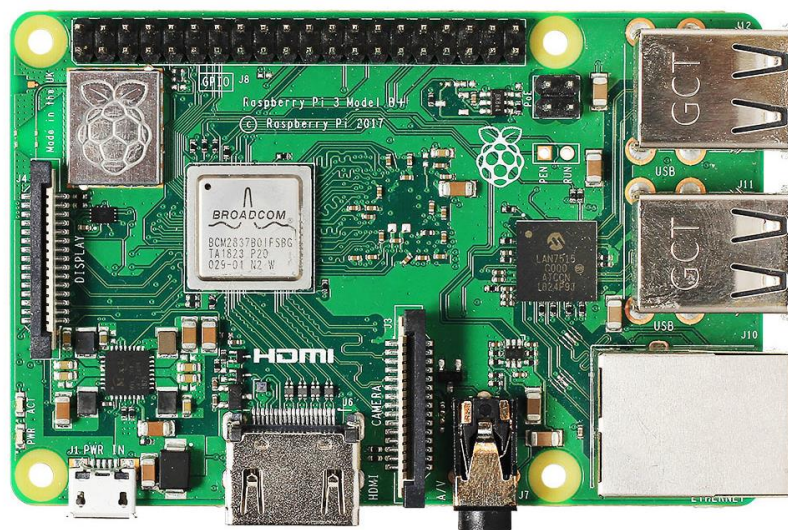


Рисунок 1.1 – Микрокомпьютер Raspberry Pi 3 B+

1.3 Подготовка к созданию системы контроля доступа

После настроек основных параметров Raspberry Pi можно приступать к разработке системы контроля доступа и учета посещаемости, однако появляется необходимость в полном наборе устройств для полноценной и удобной работы микрокомпьютера.

В проекте помимо микрокомпьютера используются RFID-считыватель и светодиод (для наглядной работы системы), требуемым напряжением питания для которых является отметка в 3.3 В.

Для подключения необходимых элементов к одноплатному компьютеру используется специально предназначенные для этого GPIO-контакты, которые следует изучить в целях правильного подключения элементов к надлежащему контакту (рисунок 1.2).

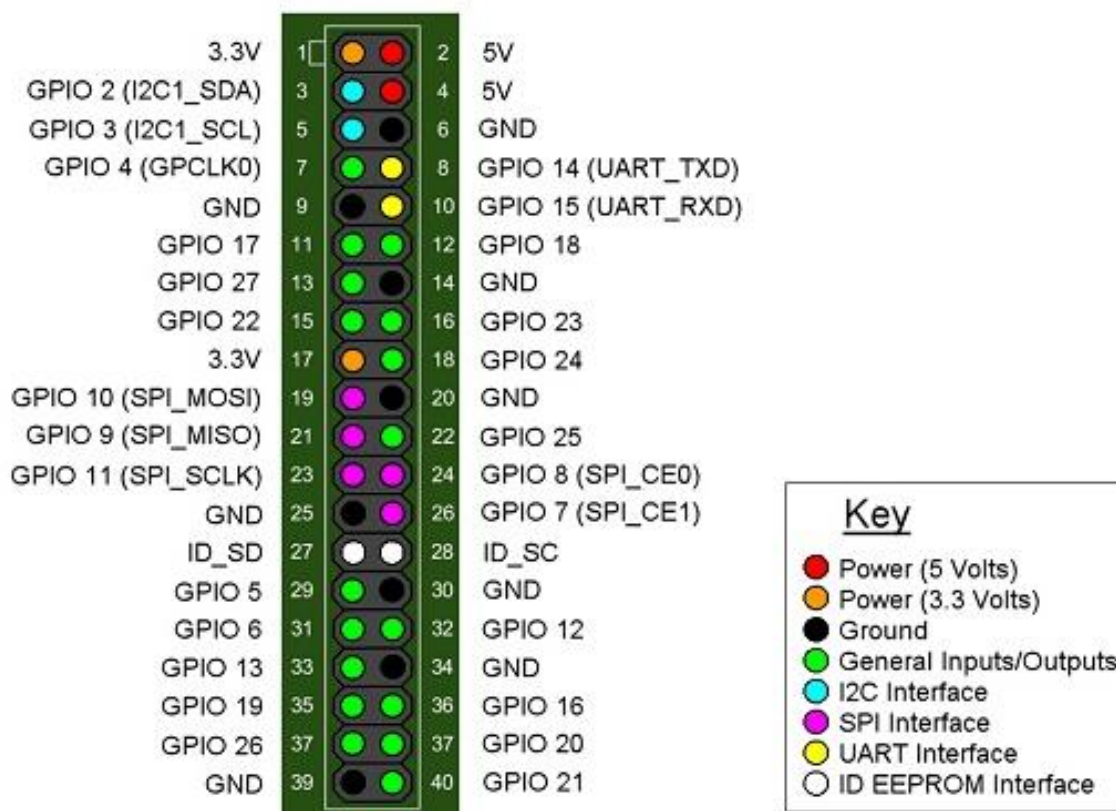


Рисунок 1.2 – Схема назначений GPIO портов

На Raspberry Pi для подключения светодиода используется контакт, предназначенный для подключения любого устройства или элемента – PIN 15, а для заземления – соответствующий этому контакту – PIN 6. В целях подключения радиочастотного идентификатора задействованы PIN 1 – питание 3.3 В, требуемое RFID считывателем, PIN 6 – заземление, PIN 19, 21, 23, 24 – контакты последовательного периферийного интерфейса, PIN 22 – контакт для сброса [2]. PIN 19, 21, 23, 24 используются для установки подключения четырех линий связи SPI:

а) PIN 19, 21 – MOSI (англ. MasterOutSlaveIn) (выход ведущего, вход ведомого; служащий для передачи данных от ведущего устройства к ведомому) и MISO (англ. MasterInSlaveOut) (вход ведущего, выход ведомого; служащий для передачи данных от ведомого к ведущему устройству) соответственно;

б) PIN 23 – SCLK или же SCK (англ. SerialClock) – последовательный тактовый сигнал, служащий для передачи тактовых сигналов для ведомых устройств;

в) PIN 24 – CE0 (англ. Chip Enable) или же чаще называют CS (англ. Chip Select) – выбор микросхемы, выбор ведомого. Подключение производится именно на этом контакте, так как ведомое устройство принимает и передает информацию, соответственно на входе ведомого устройства должен быть установлен логический ноль.

Для удобной работы и возможности разработки системы подключены все необходимые дополнительные периферийные устройства: монитор, подключенный через HDMI-кабель, либо можно использовать миниатюрный экран, поддерживающий сенсорное управление стилусом (рисунок 1.3).

Отличительным удобством данного экрана выступает тот факт, что он подсоединяется напрямую через GPIO контакты, однако тем самым создает проблему путем преграждения остальных контактов, даже имея 40 пинов, использование данного LCD исключает возможность реализации задуманной системы, так как в создании системы контроля доступа и учета посещаемости непосредственно задействованы элементы, которые необходимо подключать к микрокомпьютеру через GPIO.

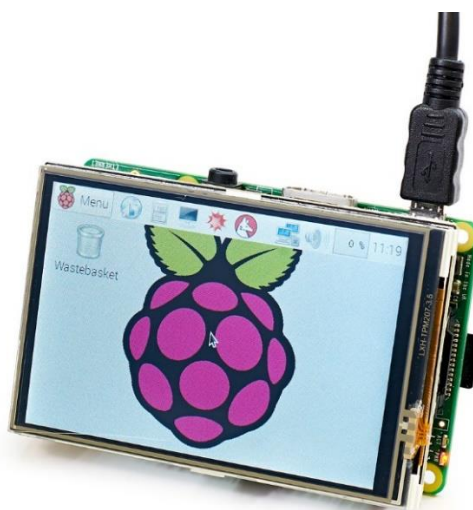


Рисунок 1.3 – ЖК-дисплей для Raspberry Pi

Заменой полноценных компьютерных мышки и клавиатуры может выступить Mini Keyboard, работающая от аккумулятора и способствующая управлению по радиоканалу (рисунок 1.4).

Её особенностями являются: наличие полной клавиатуры, touchpad, полный контроль другими устройствами, так как смартфон, смарт-ТВ, различные приборы на базе Android.

Данные решения по снабжению микрокомпьютера подключенной периферией упростило работу на данном устройстве, а также сделало рабочий

процесс возможным и, что не менее важно, сохранило свойства компактности Raspberry Pi.

В целях исключения даже минимальных финансовых затрат и видоизменения одноплатного компьютера можно воспользоваться VNC (англ. Virtual Network Computing) – системой удаленного доступа к рабочему столу компьютера (рисунок 1.5).



Рисунок 1.4 – Mini Keyboard для Raspberry Pi

Таким образом пропадает необходимость в подключении периферийных устройств, как компактных и предназначенных специально для данного микрокомпьютера, так и громоздких мониторов, клавиатур и компьютерных мышей; для подключения и настройки необходимо лишь дистанционное устройство – ноутбук, стационарный компьютер или же смартфон, программа VNC, установленная на управляющем устройстве и на Raspberry Pi, а также разрешение VNC на микрокомпьютере. Путем настройки служит создание логина и пароля для микрокомпьютера, с помощью которых в дальнейшем производится подключение к управляемой системе.

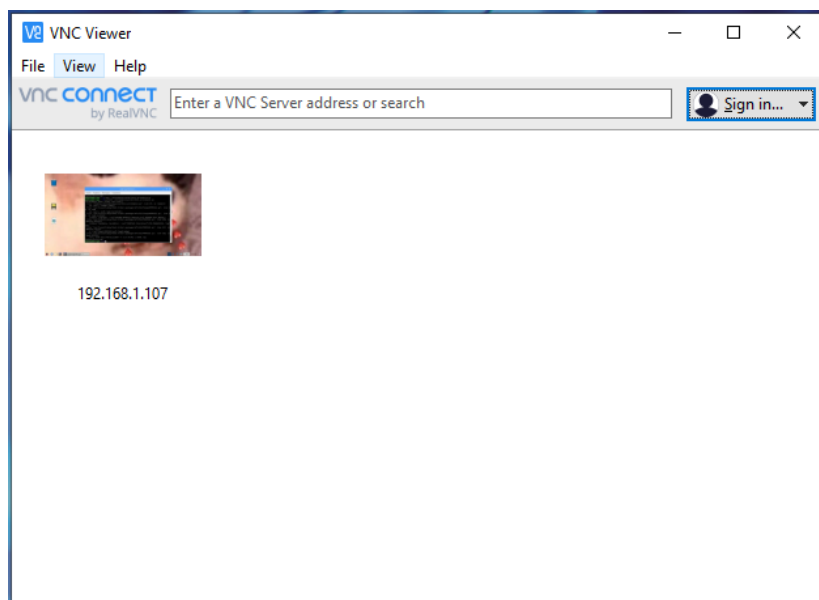


Рисунок 1.5 – Окно программы VNC Viewer

1.4 Настройка SPI

Raspberry Pi имеет шину SPI (англ. Serial Peripheral Interface - последовательный периферийный интерфейс), которую можно включить на контактах 19, 21, 23, 24 и 26. Это стандарт синхронного последовательного канала передачи данных, который используется для связи между устройствами с одним ведущим устройством на короткие расстояния. На Raspberry это обычно относится к определенным датчикам и платам расширения.

Включение данного интерфейса возможно через рабочий стол, либо путем открытия окна терминала и использовать командную строку, либо путем открытия кнопки пуска через графический инструмент «Raspberry Pi Configuration» (рисунок 1.6,1.7). Конкретное нахождение: Меню – Настройки – Raspberry Pi Configuration.

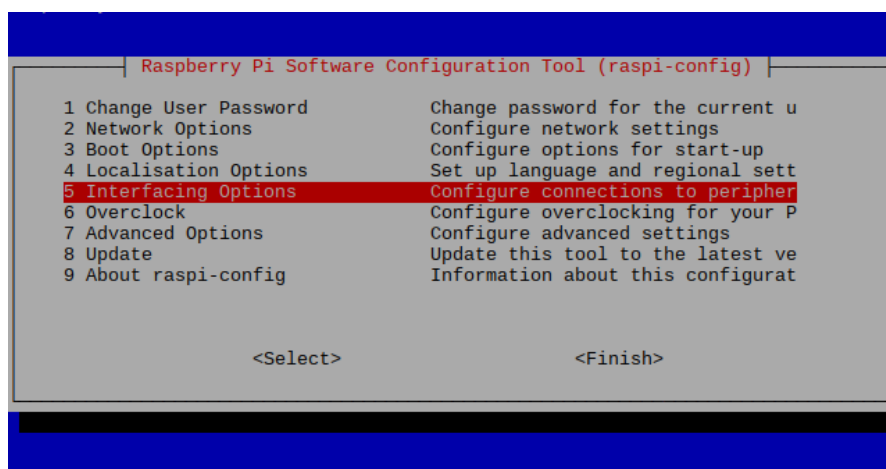


Рисунок 1.6 – Раздел конфигурации Raspberry Pi

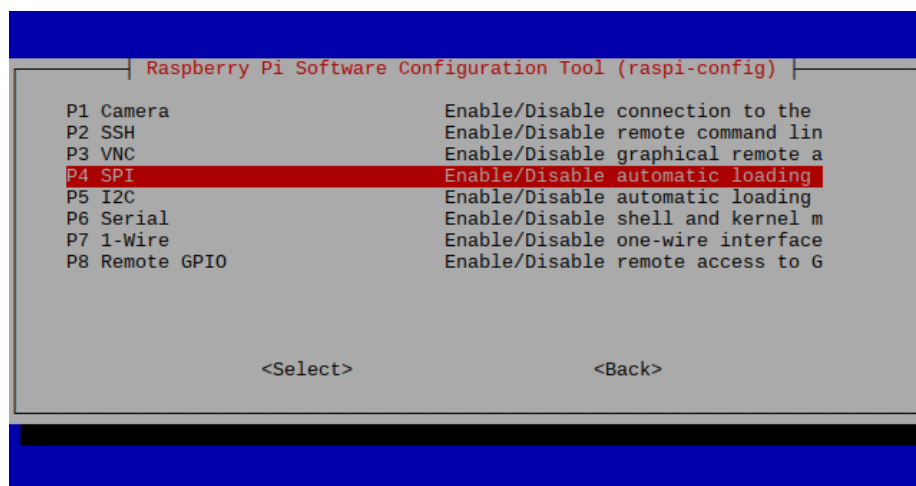


Рисунок 1.7 – Раздел настройки интерфейсов Raspberry Pi

Образ Raspbian по умолчанию отключает SPI, поэтому, прежде чем его использовать, интерфейс должен быть включен. Из командной строки или окна терминала запускается команда «`sudo raspi-config`», это задействует утилиту `raspi-config`. В открывшемся окне выбирается пункт `Interfacing Options`. Далее активируется SPI [3].

При появлении запроса на перезагрузку необходимо выбрать положительный ответ, так как только после перезагрузки Raspberry Pi интерфейс будет включен (рисунок 1.8,1.9).

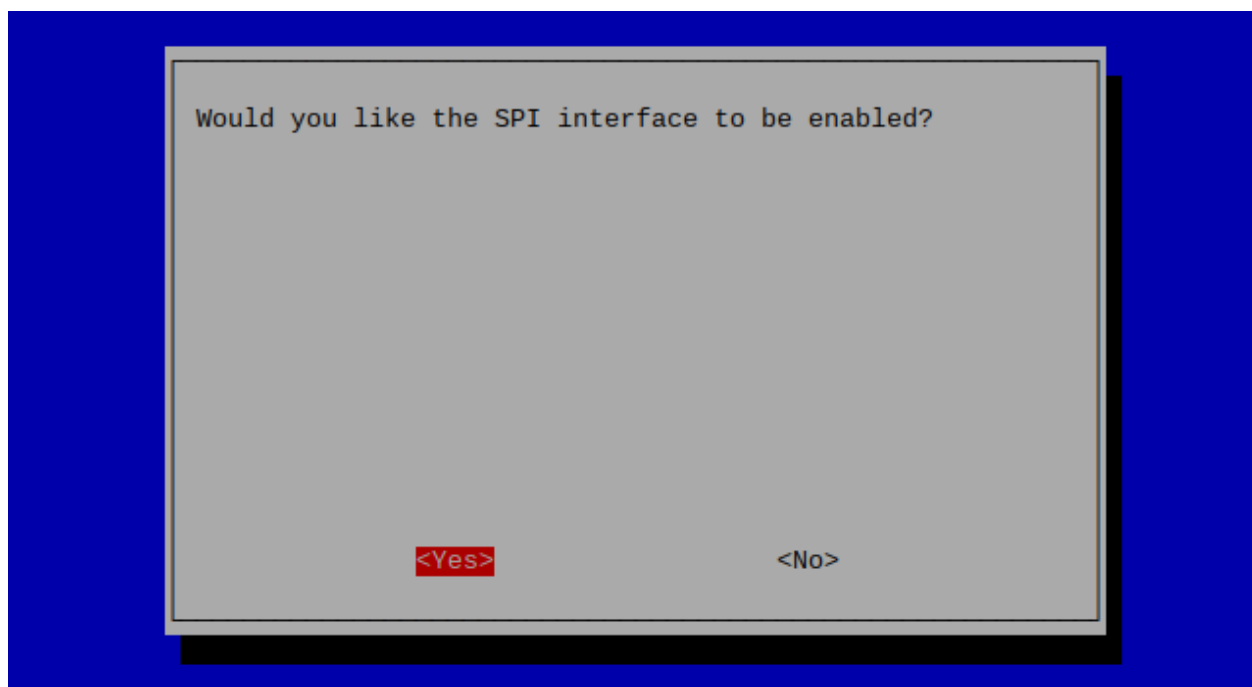


Рисунок 1.8 – Включение SPI на Raspberry Pi



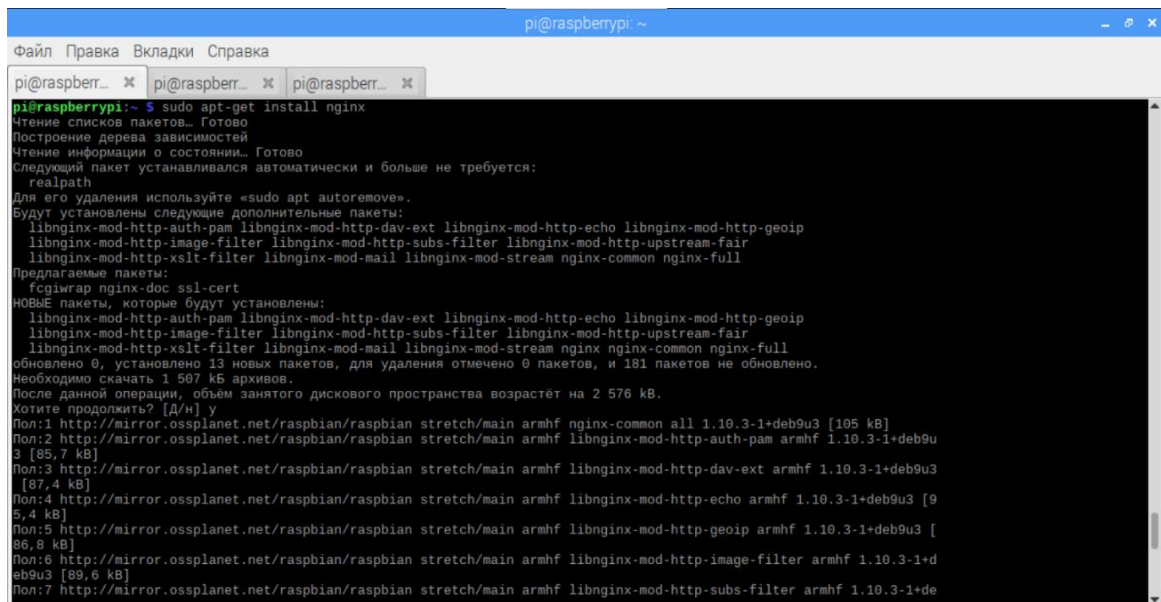
Рисунок 1.9 – Сообщение о включении SPI

1.5 Установка и настройка NGINX и PHP

NGINX - это лучшая альтернатива Apache для Raspberry Pi по причине того, что использует меньше памяти и, как правило, более низкую загрузку процессора, что означает получение больших возможностей от ограниченных ресурсов Raspberry.

Реализация данного проекта требует настроить NGINX для использования PHP-FPM, для этого требуется небольшая дополнительная настройка для запуска и работы в отличие от PHP для Apache. Хотя NGINX имеет немного больше места, чем Apache, все же не стоит ожидать, что он будет обрабатывать сколько-нибудь серьезный трафик или выполнять значительный объем работы PHP.

Прежде чем приступить к настройке веб-сервера NGINX на Raspberry Pi, нужно сначала убедиться, что Raspberry Pi обновлен, выполнив на нем две команды: «`sudo apt-get update`» и «`sudo apt-get upgrade`». Также требуется выполнение команды «`sudo apt-get remove apache2`» для удаления Apache2, так как есть вероятность, что он предварительно установлен в системе. Невыполнение этого требования может привести к сбою установки, поскольку он автоматически запускается и использует порт 80, который также использует NGINX в качестве веб-сервера. Можно пропустить этот шаг, при уверенности, что Apache2 еще не установлен на эксплуатируемом Raspberry Pi. После обновления пакетов и удаления Apache 2 можно продолжить установить NGINX, выполнив «`sudo apt-get install nginx`» (рисунок 1.10).



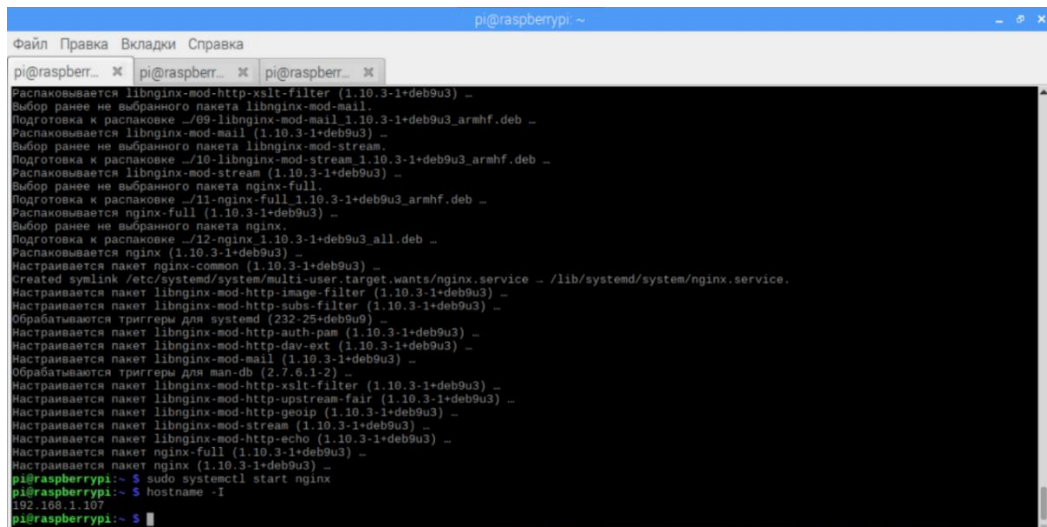
```
pi@raspberrypi: ~  
Файл Правка Вкладки Справка  
pi@raspberrypi: ~$ sudo apt-get install nginx  
Чтение списков пакетов... Готово  
Построение дерева зависимостей  
Чтение информации о состоянии... Готово  
Следующий пакет устанавливался автоматически и больше не требуется:  
  realpath  
Для его удаления используйте «sudo apt autoremove».  
Будут установлены следующие дополнительные пакеты:  
  libnginx-mod-http-auth-pam libnginx-mod-http-dav-ext libnginx-mod-http-echo libnginx-mod-http-geoip  
  libnginx-mod-http-image-filter libnginx-mod-http-subfilter libnginx-mod-http-upstream-fair  
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream nginx-common nginx-full  
Предлагаемые пакеты:  
  fcgiwrap nginx-doc ssl-cert  
НОВЫЕ пакеты, которые будут установлены:  
  libnginx-mod-http-auth-pam libnginx-mod-http-dav-ext libnginx-mod-http-echo libnginx-mod-http-geoip  
  libnginx-mod-http-image-filter libnginx-mod-http-subfilter libnginx-mod-http-upstream-fair  
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream nginx-common nginx-full  
обновлено 0, установлено 13 новых пакетов, для удаления отмечено 0 пакетов, и 181 пакетов не обновлено.  
Необходимо скачать 1 507 кБ архивов.  
После данной операции, объем занятого дискового пространства возрастет на 2 576 кБ.  
Хотите продолжить? [Д/н] y  
Пол:1 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf nginx-common all 1.10.3-1+deb9u3 [105 кБ]  
Пол:2 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf libnginx-mod-http-auth-pam armhf 1.10.3-1+deb9u3 [85,7 кБ]  
Пол:3 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf libnginx-mod-http-dav-ext armhf 1.10.3-1+deb9u3 [87,4 кБ]  
Пол:4 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf libnginx-mod-http-echo armhf 1.10.3-1+deb9u3 [95,4 кБ]  
Пол:5 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf libnginx-mod-http-geoip armhf 1.10.3-1+deb9u3 [86,8 кБ]  
Пол:6 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf libnginx-mod-http-image-filter armhf 1.10.3-1+deb9u3 [89,6 кБ]  
Пол:7 http://mirror.ossplanet.net/raspbian/raspbian stretch/main armhf libnginx-mod-http-subfilter armhf 1.10.3-1+deb9u3 [89,6 кБ]
```

Рисунок 1.10 – Отчет об установке NGINX

Теперь, когда NGINX установлен, следует запустить программное обеспечение, введя команду «sudo systemctl start nginx» в терминал, чтобы запустить веб-сервер на Raspberry Pi (рисунок 1.11).

После запуска веб-сервера NGINX нужно получить локальный IP-адрес.

Это делается для того, чтобы проверить, работает ли веб-сервер из отдельного веб-браузера. Чтобы получить локальный IP-адрес вашего Raspberry Pi используется команда «hostname -I».



```
pi@raspberrypi: ~  
Файл Правка Вкладки Справка  
pi@raspberrypi: ~$ sudo apt-get install nginx  
Распаковывается libnginx-mod-http-xslt-filter (1.10.3-1+deb9u3) ...  
Выбор ранее не выбранного пакета libnginx-mod-mail.  
Подготовка к распаковке ./09-libnginx-mod-mail_1.10.3-1+deb9u3_armhf.deb ...  
Распаковывается libnginx-mod-mail (1.10.3-1+deb9u3) ...  
Выбор ранее не выбранного пакета libnginx-mod-stream.  
Подготовка к распаковке ./10-libnginx-mod-stream_1.10.3-1+deb9u3_armhf.deb ...  
Распаковывается libnginx-mod-stream (1.10.3-1+deb9u3) ...  
Выбор ранее не выбранного пакета nginx-full.  
Подготовка к распаковке ./11-nginx-full_1.10.3-1+deb9u3_armhf.deb ...  
Распаковывается nginx-full (1.10.3-1+deb9u3) ...  
Выбор ранее не выбранного пакета nginx.  
Подготовка к распаковке ./12-nginx_1.10.3-1+deb9u3_all.deb ...  
Распаковывается nginx (1.10.3-1+deb9u3) ...  
Настраивается пакет nginx-common (1.10.3-1+deb9u3) ...  
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service -> /lib/systemd/system/nginx.service.  
Настраивается пакет libnginx-mod-http-image-filter (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-http-subfilter (1.10.3-1+deb9u3) ...  
Обрабатываются триггеры для systemd (232-25+deb9u9) ...  
Настраивается пакет libnginx-mod-http-auth-pam (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-http-dav-ext (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-mail (1.10.3-1+deb9u3) ...  
Обрабатываются триггеры для man-db (2.7.0.1-2) ...  
Настраивается пакет libnginx-mod-http-xslt-filter (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-http-upstream-fair (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-http-geoip (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-stream (1.10.3-1+deb9u3) ...  
Настраивается пакет libnginx-mod-http-echo (1.10.3-1+deb9u3) ...  
Настраивается пакет nginx-full (1.10.3-1+deb9u3) ...  
Настраивается пакет nginx (1.10.3-1+deb9u3) ...  
pi@raspberrypi: ~$ sudo systemctl start nginx  
pi@raspberrypi: ~$ hostname -I  
192.168.1.107  
pi@raspberrypi: ~$
```

Рисунок 1.11 – Запуск программного обеспечения через терминал Raspberry Pi

После запуска веб-сервера NGINX нужно получить локальный IP-адрес. Это делается для того, чтобы проверить, работает ли веб-сервер из отдельного веб-браузера. Чтобы получить локальный IP-адрес вашего Raspberry Pi используется команда «hostname -I». Теперь, когда есть

локальный IP-адрес Raspberry Pi, стоит открыть этот адрес в любом веб-браузере, просто перейдя на локальный IP-адрес, который был взят, через использование имя хоста -I. В данном случае переход совершается по адресу `http://192.168.1.107`. Как только произведен переход по адресу, видна страница с надписью «Welcome to nginx!» (рисунок 1.12).



Рисунок 1.12 – Страница приветствия NGINX

В отличие от Apache, NGINX не будет автоматически настроен для использования с PHP. Вместо этого требуется внести изменения в его конфигурационные файлы, чтобы загрузить его, также придется использовать PHP-FPM, а не стандартный PHP из-за того, как работает NGINX.

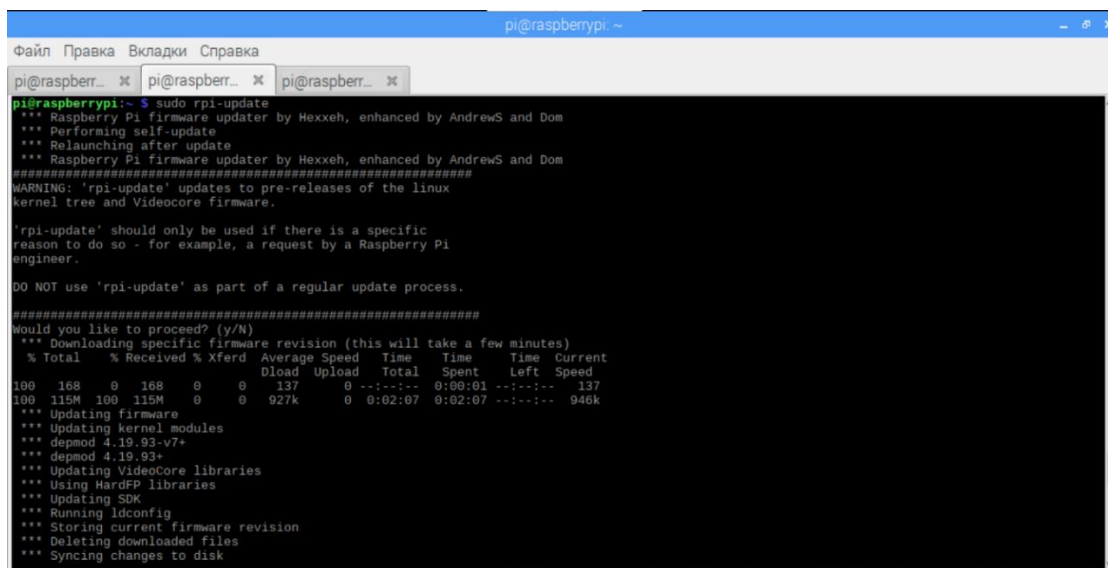
Прежде чем приступить к настройке PHP для NGINX, нужно установить PHP 7.3 и некоторые рекомендуемые модули PHP, которые облегчат работу с более обширными сценариями PHP.

Перед тем как выполнить следующую команду по установке пакетов PHP, полагается убедиться, что используется Raspbian Buster или что-то новее, в ином случае перейти с Raspbian Stretch на Buster.

Raspbian Buster - это последняя версия операционной системы Raspbian. Обновление Raspbian Buster - достаточно простой процесс, который в основном требует изменения двух простых файлов и последующего запуска обновления. Перед обновлением Raspberry Pi с Raspbian Stretch до Buster необходимо сначала сделать резервную копию SD-карты, ведь всегда есть вероятность, что этот процесс обновления сломает существующие установки на микрокомпьютере.

Если на Raspberry Pi запущено какое-то критически важное программное обеспечение, лучше всего выполнить быстрый поиск, чтобы убедиться, что в последней версии нет проблем с совместимостью. Сначала нужно убедиться, что текущая операционная система Raspbian полностью обновлена. Обновление всех установленных на данный момент пакетов гарантирует, что путь обновления до Raspbian Buster будет более чистым.

Для начала обновляются все установленные на данный момент пакеты, выполнением команд «`sudo apt update`» и «`sudo apt dist-upgrade -y`» (рисунок 1.13).



```
pi@raspberrypi:~$ sudo rpi-update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*** Performing self-update
*** Relaunching after update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*****
WARNING: 'rpi-update' updates to pre-releases of the linux
kernel tree and Videocore firmware.

'rpi-update' should only be used if there is a specific
reason to do so - for example, a request by a Raspberry Pi
engineer.

DO NOT use 'rpi-update' as part of a regular update process.

*****
Would you like to proceed? (y/n)
*** Downloading specific firmware revision (this will take a few minutes)
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
   Dload  Upload   Total   Spent    Left    Speed
100 168    0 168    0    0   137      0  0:00:01  0:00:01  0:00:00  137
100 115M 100 115M    0    0  927k      0  0:02:07  0:02:07  0:00:00 946k
*** Updating firmware
*** Updating kernel modules
*** depmod 4.19.93-v7+
*** depmod 4.19.93+
*** Updating VideoCore libraries
*** Using HardFP libraries
*** Updating SDK
*** Running ldconfig
*** Storing current firmware revision
*** Deleting downloaded files
*** Syncing changes to disk
```

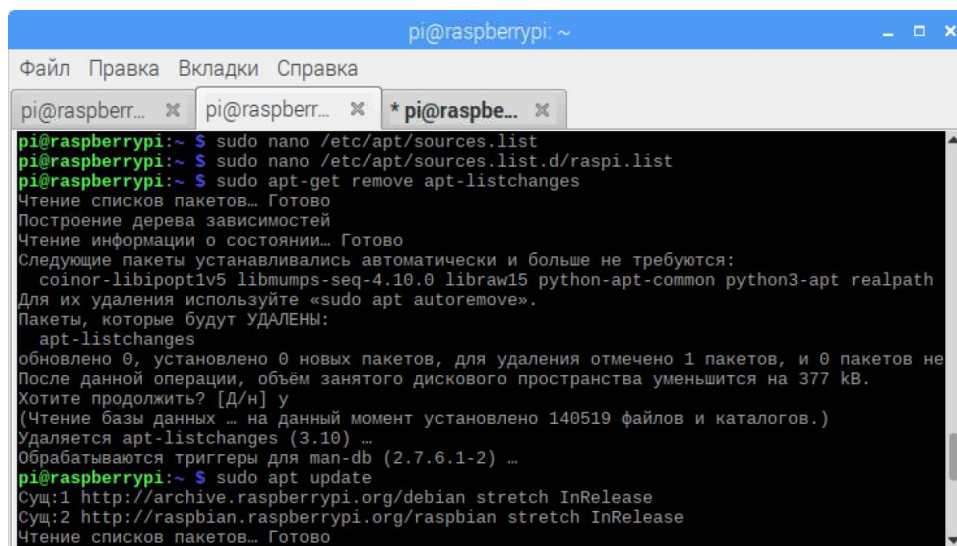
Рисунок 1.13 – Обновление прошивки Raspberry Pi

Здесь используется «`dist-upgrade`» вместо простой команды «`upgrade`», чтобы заставить Raspbian обновиться до последних доступных версий всех пакетов независимо от того, нужно ли им обновляться [4].

Далее обновляется прошивка Raspberry Pi, выполнением команды «`sudo rpi-update`».

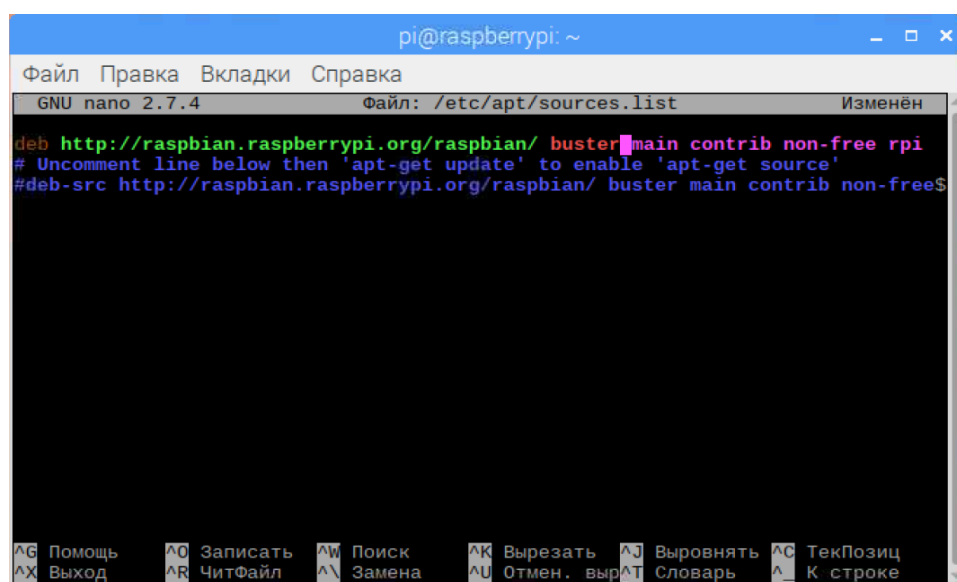
После завершения всех процессов обновления возможно перейти к обновлению Raspbian с Stretch до Raspbian Buster.

Теперь, когда установка Raspbian Stretch выполнена, можно начать процесс перехода к Buster (рисунок 1.14). Для этого необходимо изменить файл «`/etc/apt/sources.list`», выполнив «`sudo nano /etc/apt/sources.list`» (рисунок 1.15).



```
pi@raspberrypi:~$ sudo nano /etc/apt/sources.list
pi@raspberrypi:~$ sudo nano /etc/apt/sources.list.d/raspi.list
pi@raspberrypi:~$ sudo apt-get remove apt-listchanges
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Следующие пакеты устанавливались автоматически и больше не требуются:
  coinor-libiortlv5 libtumps-seq-4.10.0 libraw15 python-apt-common python3-apt realpath
Для их удаления используйте «sudo apt autoremove».
Пакеты, которые будут УДАЛЕНЫ:
  apt-listchanges
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 1 пакетов, и 0 пакетов не
После данной операции, объём занятого дискового пространства уменьшится на 377 kB.
Хотите продолжить? [Д/н] y
(Чтение базы данных ... на данный момент установлено 140519 файлов и каталогов.)
Удаляется apt-listchanges (3.10) ...
Обрабатываются триггеры для man-db (2.7.6.1-2) ...
pi@raspberrypi:~$ sudo apt update
Сущ:1 http://archive.raspberrypi.org/debian stretch InRelease
Сущ:2 http://raspbian.raspberrypi.org/raspbian stretch InRelease
Чтение списков пакетов... Готово
```

Рисунок 1.14 – Окно терминала с вводом нескольких команд



```
GNU nano 2.7.4 Файл: /etc/apt/sources.list Изменён
deb http://raspbian.raspberrypi.org/raspbian/ buster main contrib non-free rpi
# Uncomment line below then 'apt-get update' to enable 'apt-get source'
#deb-src http://raspbian.raspberrypi.org/raspbian/ buster main contrib non-free$
```

Рисунок 1.15 – Изменение файла «/etc/apt/sources.list»

В этом файле «Stretch» изменяется на «Buster». Это изменение позволит менеджеру пакетов выполнять поиск в хранилище пакетов Raspberry Pi в дистрибутиве «Buster» вместо дистрибутива «Stretch».

Изменить «deb http://raspbian.raspberrypi.org/raspbian/ stretch main contrib non-free rpi» на «deb http://raspbian.raspberrypi.org/raspbian/ buster main contrib non-free rpi».

Как только были произведены изменения в файле, происходит сохранение, нажатием клавиш CTRL + X, затем Y, а затем ENTER .

После нужно изменить файл «/etc/apt/sources.list.d/raspi.list», выполнив команду «sudo nano /etc/apt/sources.list.d/raspi.list».

В этом файле, аналогично первому, «Stretch» изменяется на «Buster», а строка «deb http://archive.raspberrypi.org/debian/ stretch main» на «deb http://archive.raspberrypi.org/debian/ buster main» (рисунок 1.16).

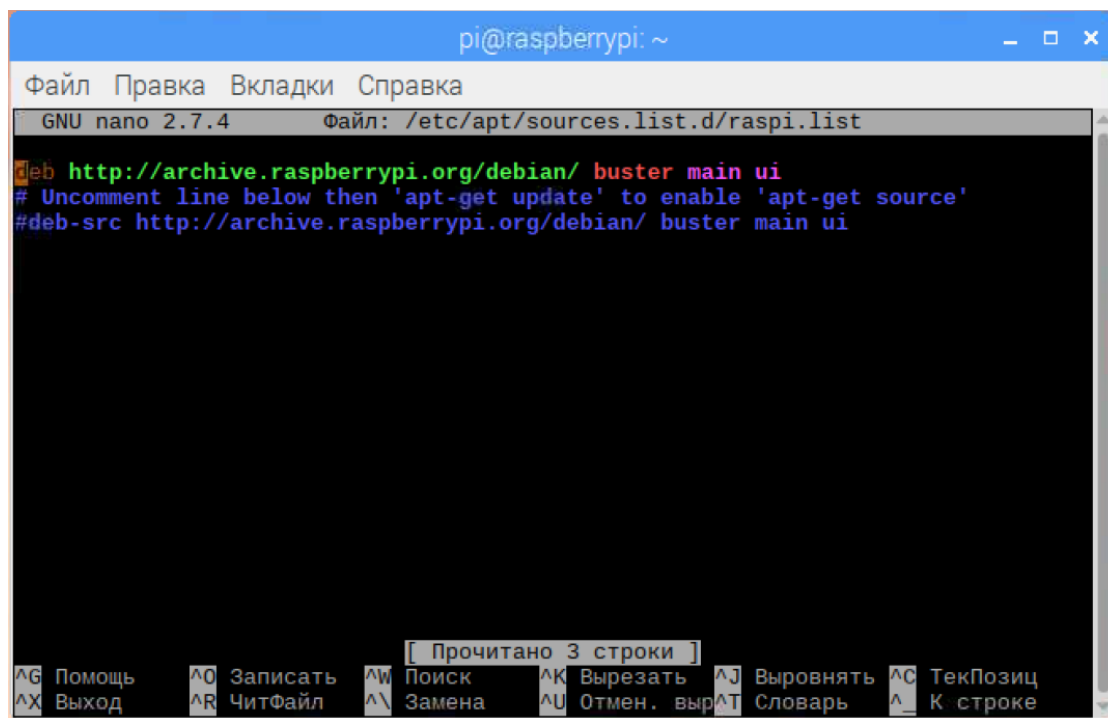


Рисунок 1.16 – Изменение файла «/etc/apt/sources.list.d/raspi.list»

Сохраняется файл тем же способом.

Теперь, прежде чем сделать последний шаг к Raspbian Buster, надо удалить пакет «apt-listchanges».

Причина удаления этого пакета состоит в том, чтобы ускорить процесс обновления, без удаления этого пакета операционная система Raspbian должна будет загрузить довольно большой файл журнала изменений, который значительно замедлит процесс обновления. Удаляется пакет «apt-listchanges» командой «sudo apt-get remove apt-listchanges».

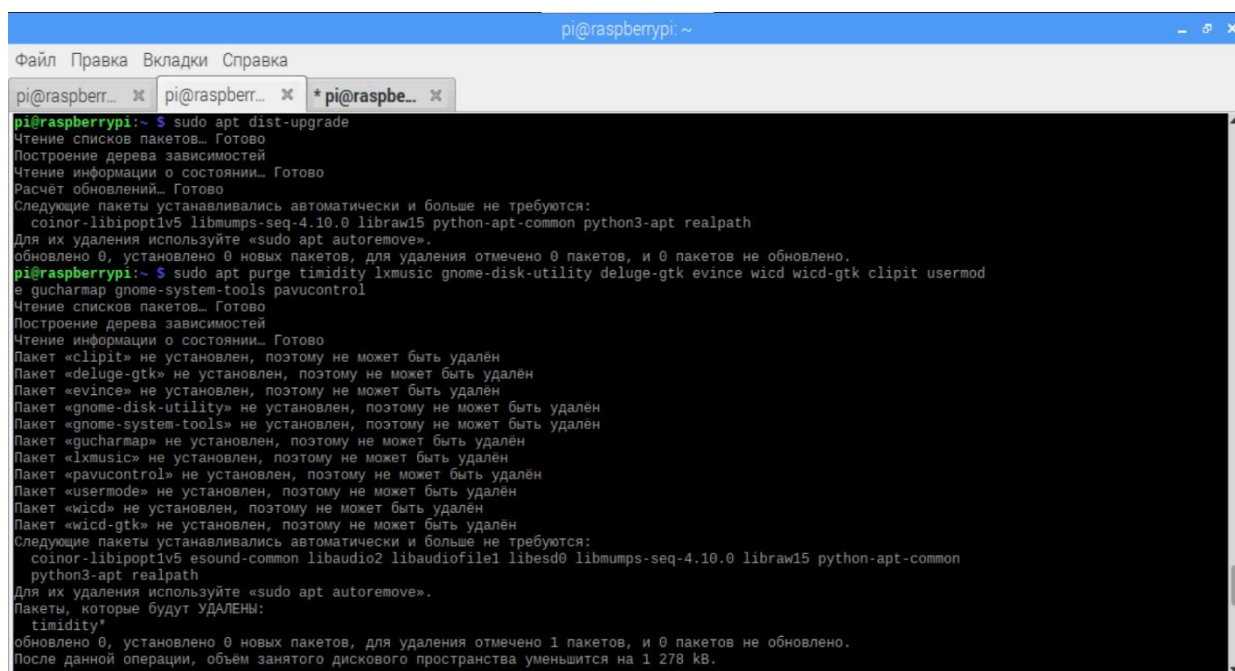
Наконец, теперь, исходные файлы изменены, что позволяет установить сборку «Buster» вместо «Stretch».

Первая команда «sudo apt update» обновит списки пакетов, хранящиеся на Raspberry Pi, вторая команда «sudo apt dist-upgrade» обновит все пакеты до их версий Raspbian Buster.

Этот процесс может занять значительное время, так как существует довольно много пакетов, которые необходимо обновить, кроме того, может потребоваться отвечать на запросы.

После завершения процесса обновления Buster требуется избавиться от некоторых новых приложений, которые будут автоматически установлены, эти пакеты не поддерживаются Raspberry Pi и их рекомендуется удалить командой «sudo apt purge timidity lxmusic gnome-disk-utility deluge-gtk evince

wicd wicd-gtk clipit usermode gucharmap gnome-system-tools pavucontrol»
(рисунок 1.17).



```
pi@raspberrypi: ~  
Файл Правка Вкладки Справка  
pi@raspberrypi:~$ sudo apt dist-upgrade  
Чтение списков пакетов... Готово  
Построение дерева зависимостей  
Чтение информации о состоянии... Готово  
Расчет обновлений... Готово  
Следующие пакеты устанавливались автоматически и больше не требуются:  
coinor-libipopt1v5 libmumps-seq-4.10.0 libraw15 python-apt-common python3-apt realpath  
Для их удаления используйте «sudo apt autoremove».  
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.  
pi@raspberrypi:~$ sudo apt purge timidity lxmusic gnome-disk-utility deluge-gtk evince wicd wicd-gtk clipit usermod  
e gucharmap gnome-system-tools pavucontrol  
Чтение списков пакетов... Готово  
Построение дерева зависимостей  
Чтение информации о состоянии... Готово  
Пакет «clipit» не установлен, поэтому не может быть удален  
Пакет «deluge-gtk» не установлен, поэтому не может быть удален  
Пакет «evince» не установлен, поэтому не может быть удален  
Пакет «gnome-disk-utility» не установлен, поэтому не может быть удален  
Пакет «gnome-system-tools» не установлен, поэтому не может быть удален  
Пакет «gucharmap» не установлен, поэтому не может быть удален  
Пакет «lxmusic» не установлен, поэтому не может быть удален  
Пакет «pavucontrol» не установлен, поэтому не может быть удален  
Пакет «usermode» не установлен, поэтому не может быть удален  
Пакет «wicd» не установлен, поэтому не может быть удален  
Пакет «wicd-gtk» не установлен, поэтому не может быть удален  
Следующие пакеты устанавливались автоматически и больше не требуются:  
coinor-libipopt1v5 esound-common libaudio2 libaudiofile1 libesd0 libmumps-seq-4.10.0 libraw15 python-apt-common  
python3-apt realpath  
Для их удаления используйте «sudo apt autoremove».  
Пакеты, которые будут УДАЛЕНЫ:  
timidity*  
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 1 пакетов, и 0 пакетов не обновлено.  
После данной операции, объем занятого дискового пространства уменьшится на 1 278 kB.
```

Рисунок 1.17 – Отчет об удалении ненужных пакетов

После нужно выполнить еще несколько команд, чтобы убедиться, что очистилось все, что осталось от обновления.

Первая команда «`sudo apt autoremove -y`» удалит все пакеты, которые были помечены как ненужные из-за измененных зависимостей.

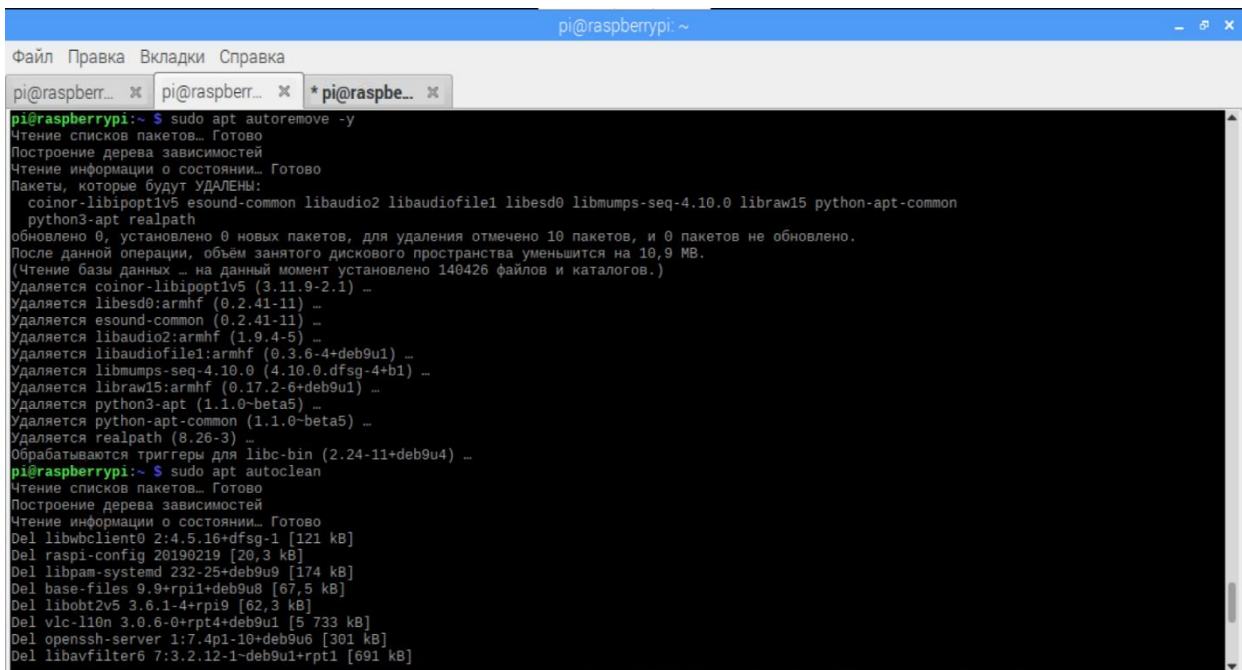
Следующая команда «`sudo apt autoclean`» очистит кэш пакета (рисунок 1.18). Она автоматически удаляет все файлы пакета, которые больше не доступны для загрузки и, следовательно, в основном бесполезны.

Последнее, что полагается сделать - перезагрузить Raspberry Pi командой «`sudo reboot`». Перезапуск гарантирует, что Raspberry Pi загрузит все новые пакеты Buster и очистит все старые данные, хранящиеся в памяти.

К этому моменту Raspberry Pi успешно обновлен с Raspbian Stretch до Raspbian Buster. Теперь можно продолжить подготовку к разработке проекта, используя обновленную версию Raspbian.

Далее следует установка PHP пакетов командой «`sudo apt-get install php7.3-fpm php7.3-mbstring php7.3-mysql php7.3-curl php7.3-gd php7.3-curl php7.3-zip php7.3-xml -y`».

С установленным PHP-FPM можно внести необходимые изменения в файл конфигурации NGINX по умолчанию. Чтобы начать редактирование этого файла, используется команда «`sudo nano /etc/nginx/sites-enabled/default`».



```
pi@raspberrypi: ~  
Файл Правка Вкладки Справка  
pi@raspberrypi:~$ sudo apt autoremove -y  
Чтение списков пакетов... Готово  
Построение дерева зависимостей  
Чтение информации о состоянии... Готово  
Пакеты, которые будут УДАЛЕНЫ:  
  coignor-libiport1v5 esound-common libaudio2 libaudiofile1 libesd0 libmumps-seq-4.10.0 libraw15 python-apt-common  
  python3-apt realpath  
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 10 пакетов, и 0 пакетов не обновлено.  
После данной операции, объем занятого дискового пространства уменьшится на 10,9 MB.  
(Чтение базы данных ... на данный момент установлено 140426 файлов и каталогов.)  
Удаляется coignor-libiport1v5 (3.11.9-2.1) ...  
Удаляется libesd0:armhf (0.2.41-11) ...  
Удаляется esound-common (0.2.41-11) ...  
Удаляется libaudio2:armhf (1.9.4-5) ...  
Удаляется libaudiofile1:armhf (0.3.6-4+deb9u1) ...  
Удаляется libmumps-seq-4.10.0 (4.10.0.dfsg-4+b1) ...  
Удаляется libraw15:armhf (0.17.2-6+deb9u1) ...  
Удаляется python3-apt (1.1.0-beta5) ...  
Удаляется python-apt-common (1.1.0-beta5) ...  
Удаляется realpath (8.26-3) ...  
Обрабатываются триггеры для libc-bin (2.24-11+deb9u4) ...  
pi@raspberrypi:~$ sudo apt autoclean  
Чтение списков пакетов... Готово  
Построение дерева зависимостей  
Чтение информации о состоянии... Готово  
Del libwbclient0 2:4.5.16+dfsg-1 [121 kB]  
Del raspi-config 20190219 [20,3 kB]  
Del libram-systemd 232-25+deb9u9 [174 kB]  
Del base-files 9.9+rp11+deb9u8 [67,5 kB]  
Del libobt2v5 3.6.1-4+rp19 [62,3 kB]  
Del vlc-l10n 3.0.6-0+rp14+deb9u1 [5 733 kB]  
Del openssh-server 1:7.4p1-10+deb9u6 [301 kB]  
Del libavfilter6 7:3.2.12-1+deb9u1+rp11 [691 kB]
```

Рисунок 1.18 – Автоматическая очистка кэша пакета

В этом файле требуется найти «index index.html index.htm;» и заменить на «index index.php index.html index.htm;». Здесь также нужно добавить «index.php» в строку индекса - это говорит NGINX распознавать файл «index.php» как возможный индекс, а добавление его первым в списке означает, что он будет выбран над файлом «index.html» (рисунок 1.19).

Найти:

```
#location ~ \.php$ {  
    #    include snippets/fastcgi-php.conf;  
    #  
    #    # With php5-cgi alone:  
    #    fastcgi_pass 127.0.0.1:9000;  
    #    # With php5-fpm:  
    #    fastcgi_pass unix:/var/run/php5-fpm.sock;  
    #}
```

Заменить на:

```
location ~ \.php$ {  
    include snippets/fastcgi-php.conf;  
    fastcgi_pass unix:/var/run/php/php7.3-fpm.sock;  
    }  
}
```

Этот код настраивает NGINX для обработки файлов .php, передавая их в PHP-FPM.

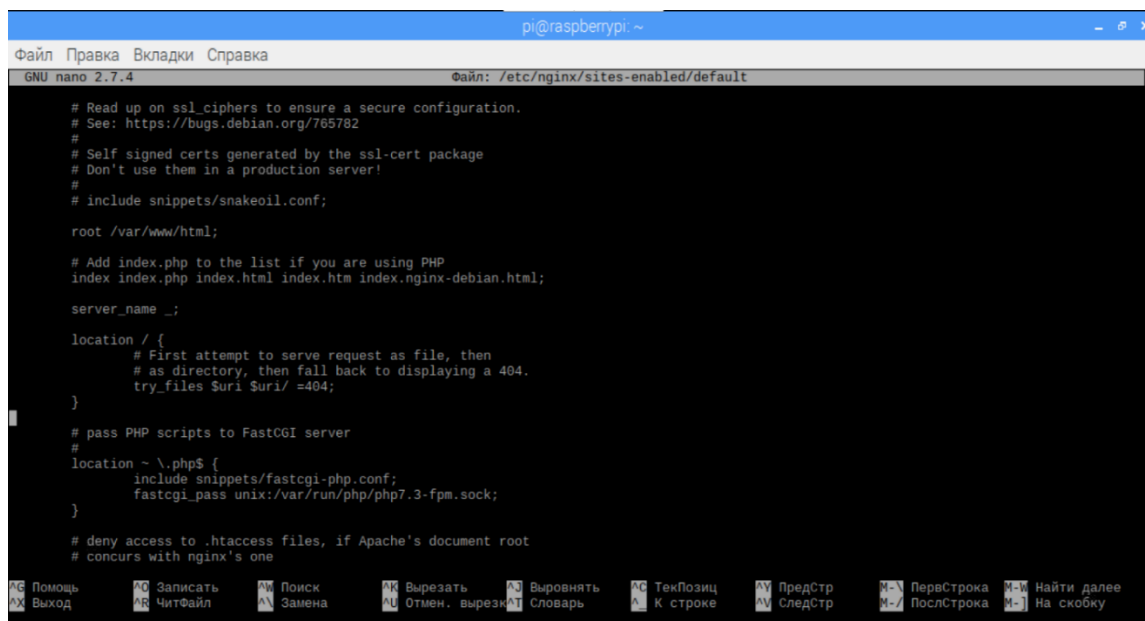


Рисунок 1.19 – Изменение кода

Далее нужно указать NGINX перезагрузить конфигурацию, выполнив команду «sudo systemctl reload nginx» (рисунок1.20).

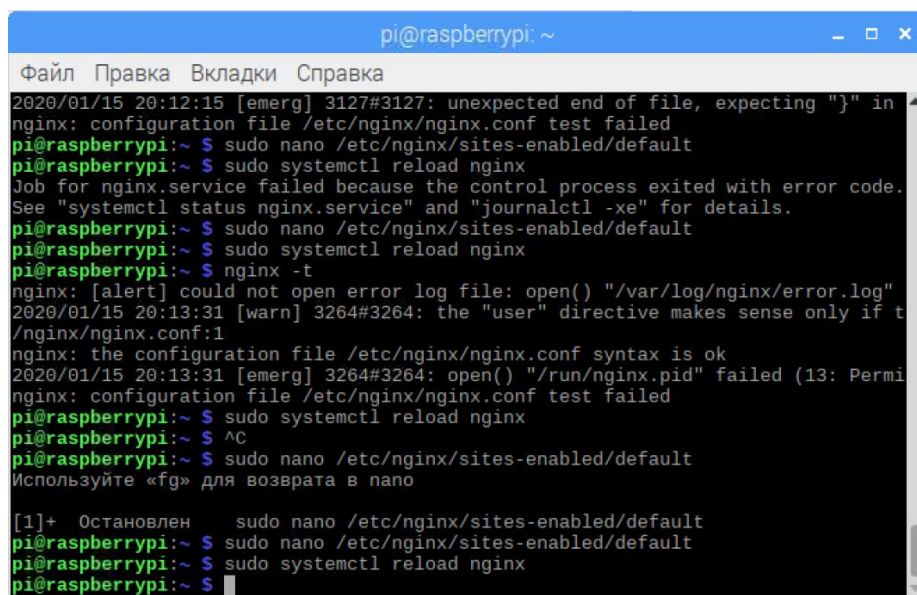


Рисунок 1.20 – Перезагрузка конфигурации NGINX

Наконец, тест установки PHP. Написав очень простой файл «index.php» в каталоге «/var/www/html», запускается команда «sudo nano /var/www/html/index.php», чтобы создать и начать редактирование файла «index.php». В этот файл добавляется строка «<?php phpinfo(); ?>».

Теперь, как и ранее производится переход на IP-адрес Raspberry Pi, но на этот раз открывается страница со всей информацией о версии PHP и о том,

какие модули активны в данный момент (рисунок 1.21). Это признак того, что установка PHP запущена и работает правильно.



Рисунок 1.21 – Информационное окно PHP

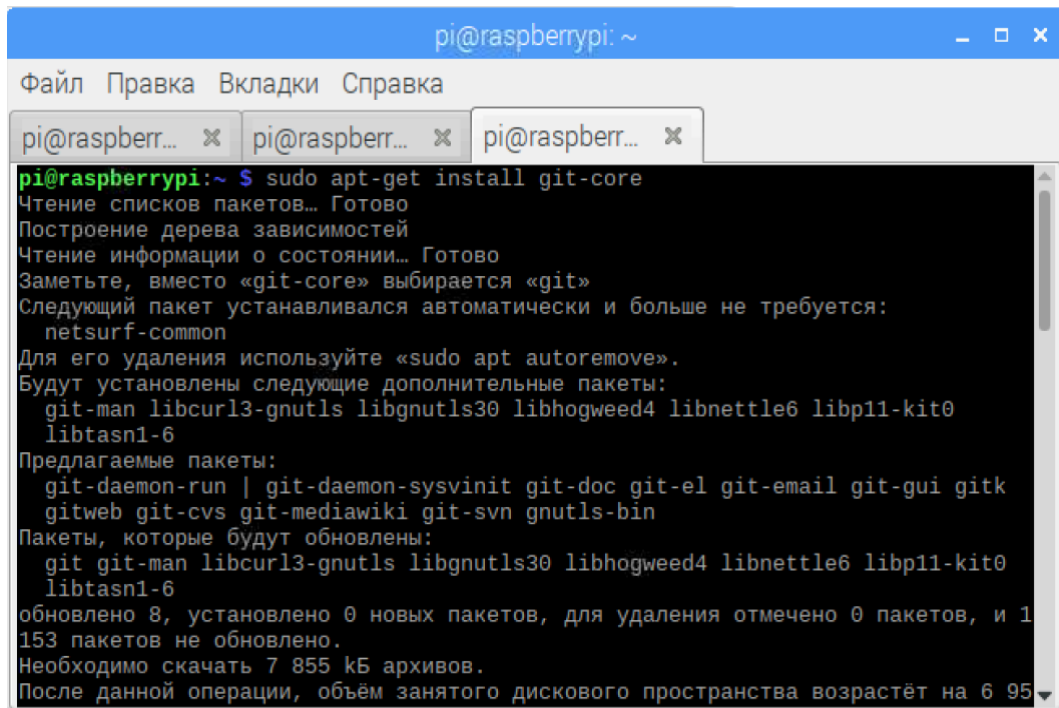
1.6 Подключение и настройка Git и SSH

Git – популярная версия программного обеспечения, служащее для контроля разработки программного обеспечения. Существует множество размещенных Git-решений, но, возможно, в некоторых ситуациях нужен собственный Git-сервер по той или иной причине. Например, полный контроль над созданным кодом и возможность сохранять его как можно более приватным.

Преимущество изучения создания собственного Git-сервера состоит в том, что пользователь получает представление о большинстве команд Git. Знание этих команд чрезвычайно удобно, особенно в тех проектах, когда не используют графический интерфейс GIT.

Процесс установки Git сервера на Raspberry pi довольно прост. Как и прежде, сначала необходимо удостовериться, что Pi обновлен, с помощью команд «sudo apt-get update» и «sudo apt-get upgrade». Далее требуется убедиться, что Git установлен через команду «sudo apt-get install git-core» (рисунок 1.22).

Следующим важным шагом выступает включение SSH. SSH (англ. Secure Shell) представляет собой протокол, обеспечивающий соединение по сети и его поддержку (интернет или локальной сети) между двумя удаленными устройствами и является одним из самых полезных способов для удаленного управления устройством, таким как Raspberry Pi. SSH дает возможность передавать практически любую информацию (например, файлы, аудио и видео в потоке и так далее).



```
pi@raspberrypi: ~
Файл Правка Вкладки Справка
pi@raspberr... x pi@raspberr... x pi@raspberr... x
pi@raspberrypi:~ $ sudo apt-get install git-core
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Заметьте, вместо «git-core» выбирается «git»
Следующий пакет устанавливался автоматически и больше не требуется:
  netsurf-common
Для его удаления используйте «sudo apt autoremove».
Будут установлены следующие дополнительные пакеты:
  git-man libcurl3-gnutls libgnutls30 libhogweed4 libnettle6 libp11-kit0
  libtasn1-6
Предлагаемые пакеты:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-cvs git-mediawiki git-svn gnutls-bin
Пакеты, которые будут обновлены:
  git git-man libcurl3-gnutls libgnutls30 libhogweed4 libnettle6 libp11-kit0
  libtasn1-6
обновлено 8, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 1
153 пакетов не обновлено.
Необходимо скачать 7 855 кБ архивов.
После данной операции, объем занятого дискового пространства возрастёт на 6 95
```

Рисунок 1.22 – Установка Git

Главными преимуществами данного протокола являются: во-первых, умение сжимать данные на стадии передачи; во-вторых, с целью обеспечения безопасности соединения, создание зашифрованного туннеля.

Итак, весь этот процесс - все, что нужно для установки и запуска сервера Git.

2 Разработка системы учета посещаемости

2.1 Постановка задачи

Полная подготовка необходимых для разработки составляющих, а также настройка основных подключений представляют наполовину выполненную реализацию системы контроля доступа и позволяют перейти к непосредственному созданию действующих рычагов проекта.

Одной из самых простых и элементарных задач выступает управление светодиодом, то есть, регулирование его включения и выключения, возможность учета и установки времени для данных процессов (рисунок 2.1). В функционирование данного проекта заложено подключение светодиода к одноплатному компьютеру Raspberry Pi путем написания соответствующей программы на языке Python, которая позволит управлять данным элементом и заставлять его работать.

Ранее, при рассмотрении технических характеристик Raspberry Pi, упоминались некоторые преимущества данного устройства, а конкретно – наличие входов-выводов общего назначения, так называемых GPIO (General Purpose Input/Output). GPIO представляют собой некую группу контактов, представляющих управлять ими при помощи программы, к примеру, светодиодом, где реализация данного процесса является легкой, или же функционирование более сложных устройств, требующих использование некоторых специальных протоколов для передачи данных между этими устройствами.

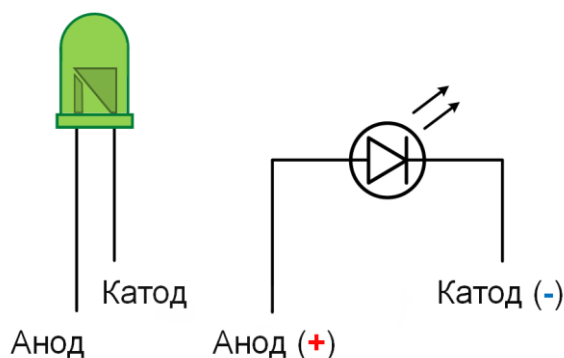


Рисунок 2.1 – Параметры светодиода

Работа контактов состоит в возможности соединения необходимого контакта с контактом питания на 3.3 В или с заземлением (GND). Иными словами, при подключении устройства к выбранному контакту, по написанной программе при подаче на вывод значения истины/true Raspberry соединит данный вывод с питанием, таким образом через подключенное к выбранному контакту устройство будет идти ток.

В случае подачи значения ложь/false произойдет соединение с заземлением, что означает отсутствие питания. Данные манипуляции

представляют собой своеобразное переключение состояния устройства наподобие переключателя света.

Светодиод представляет собой полупроводниковое устройство с электронно-дырочным переходом, которое излучает свет во время пропускания через него электрического тока в одном направлении (от анода к катоду).

В целях правильного подключения следует отличать анод от катода: анод светодиода имеет проводник длиннее, чем у катода.

RFID радиочастотная идентификация представляет собой устройство для бесконтактной идентификации объектов, которое использует радиочастотный канал связи (рисунок 2.2). Анализ объектов происходит по уникальному идентификатору, имеющийся в электронной метке каждой из карт. Функционирование данного устройство представляет собой процесс, при котором метки отправляют информацию, полученную идентификатором, в ответ на излучаемые считывателем электромагнитные волны определенной частоты [5].

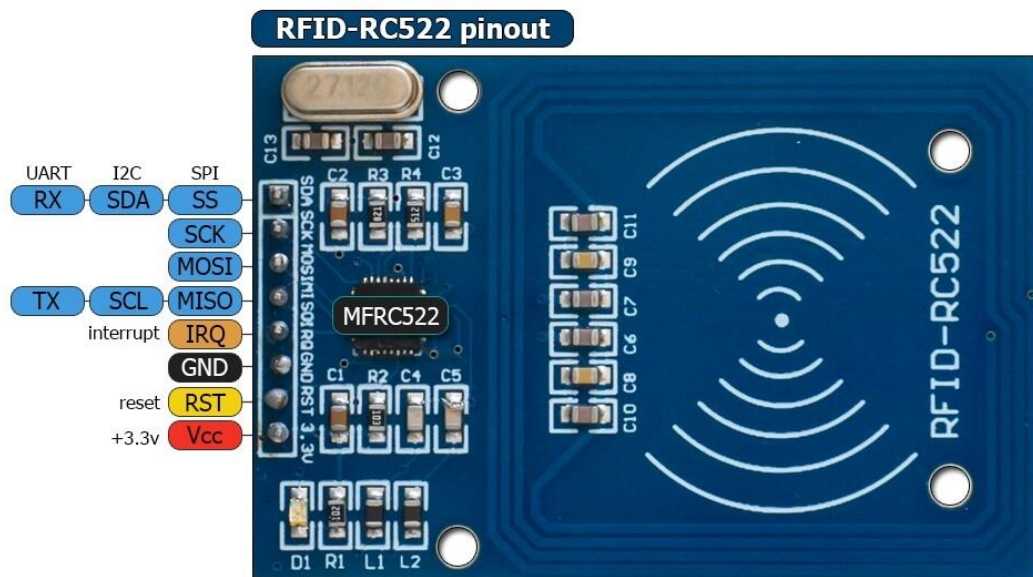


Рисунок 2.2 – Свойства RFID-считывателя

Микросхема, на которой выполнен модуль, MFRC522 поддерживает следующие интерфейсы: SPI, I2C, UART. Для выбора интерфейса необходимо установить логические уровни на определенные выводы микросхемы, так как на модуле выбран SPI.

Выводы SPI:

- а) SDA – выбор ведомого;
- б) SCK – сигнал синхронизации;
- в) MOSI – от ведущего к ведомому;
- г) MISO – от ведомого к ведущему;
- д) IRQ – прерывание;

- е) GND – заземление;
- ж) RST – сброс;
- з) Vcc – питание на 3.3 В.

2.2 Общее соединение и вид устройства

Внешний вид макета и общее соединение для системы контроля доступа и учета посещаемости показан на рисунке 2.3.

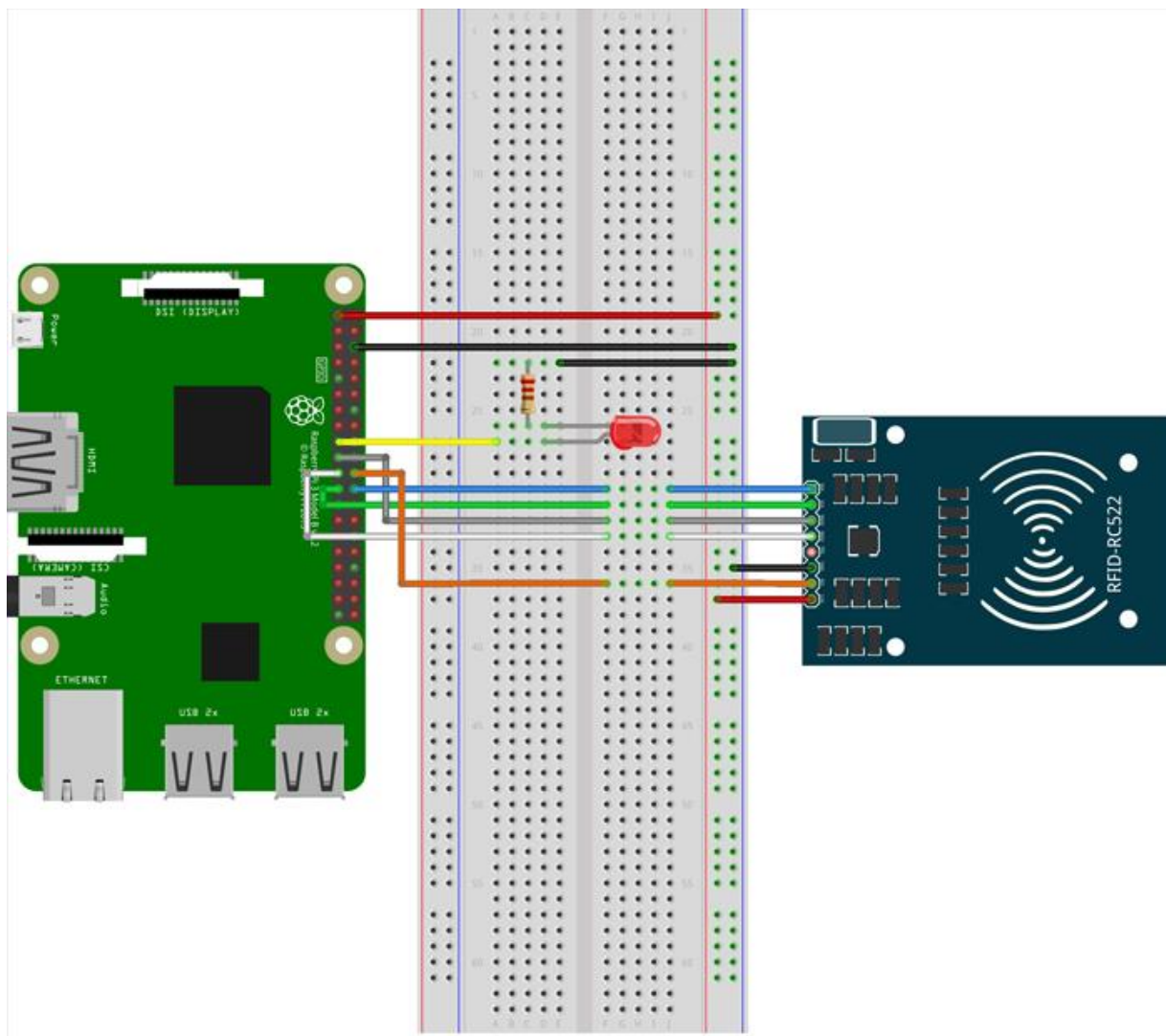


Рисунок 2.3 – Вид макета и соединения

Кроме светодиода и считывателя в данной схеме имеется резистор, необходимый для правильной работы светодиода и во избежание его перегорания, так как резистор выдерживает рабочее напряжение и ток и позволяет установить их соответствующие показатели для исправной работы.

Для установки нужного резистора с необходимым номиналом можно воспользоваться законом Ома для участка цепи – сила тока на участке цепи

прямо пропорциональна напряжению и обратно пропорциональна сопротивлению участка цепи:

$$I = U/R, \quad (1.1)$$

где I – сила тока, А;

U – напряжение, В;

R – сопротивление, Ом.

Можно выразить из этой формулы сопротивление R :

$$R = U/I. \quad (1.2)$$

В данном проекте используется зеленый светодиод, рабочий ток которого составляет 20 мА при напряжении от 1.8 до 2 В.

Так как Raspberry Pi имеет напряжение в 3.3 В, необходимо отсечь лишние 1.3 В, поэтому сопротивление резистора при токе 20 мА, чтобы избавиться от 1.3 В, должно составлять:

$$R = 1,3\text{В}/20\text{мА} = 65 \text{ Ом}.$$

При отсутствии резистора на 65 Ом, можно воспользоваться резистором другого большего номинала, к примеру, 100 Ом или же 200 Ом, но при этом светодиод будет гореть тусклее.

2.3 Создание схемы считывателя RFID RC522

Для начала следует убедиться, что в установленной операционной системе Raspbian все обновлено, выполнив команды `sudo apt-get update` и `sudo apt-get upgrade`.

Теперь нужно установить все пакеты, которые послужат опорой для последующей разработки системы. Начать можно с установки `build-essential`, `python3-dev`, `python3-pip` и `python3-smbus`, выполнив команду `sudo apt-get install build-essential git python3-dev python3-pip python3-smbus`.

Для подключения RFID RC522 к цепи понадобится:

- а) 1 соединительный провод папа-папа;
- б) 6 соединительных проводов папа-мама;
- в) RFID RC522 для чтения и записи;
- г) макетная плата.

Подключение RFID к Raspberry Pi происходит следующим образом (рисунок 2.4):

- а) SDA подключается к GPIO8 (физический вывод 24);
- б) SCK подключается к GPIO11 (физический вывод 23);
- в) MOSI подключается к GPIO10 (физический контакт 19);
- г) MISO подключается к GPIO9 (физический контакт 21);
- д) GND соединяется с подключенным к шине заземления на макете физическим контактом 6 (GND);
- е) RST подключается к GPIO25 (физический контакт 22);
- ж) 3.3v подключается к 3.3v (физический контакт 1).

Теперь, когда RFID подключен к Raspberry Pi, убедиться, что интерфейс SPI включен, выполнив команду «`lsmod | grep spi`». Эта команда извлекает список включенных модулей ядра и извлекает из этого списка все, что содержит текст «spi». Интерфейс необходим для связи с модулем RC522.

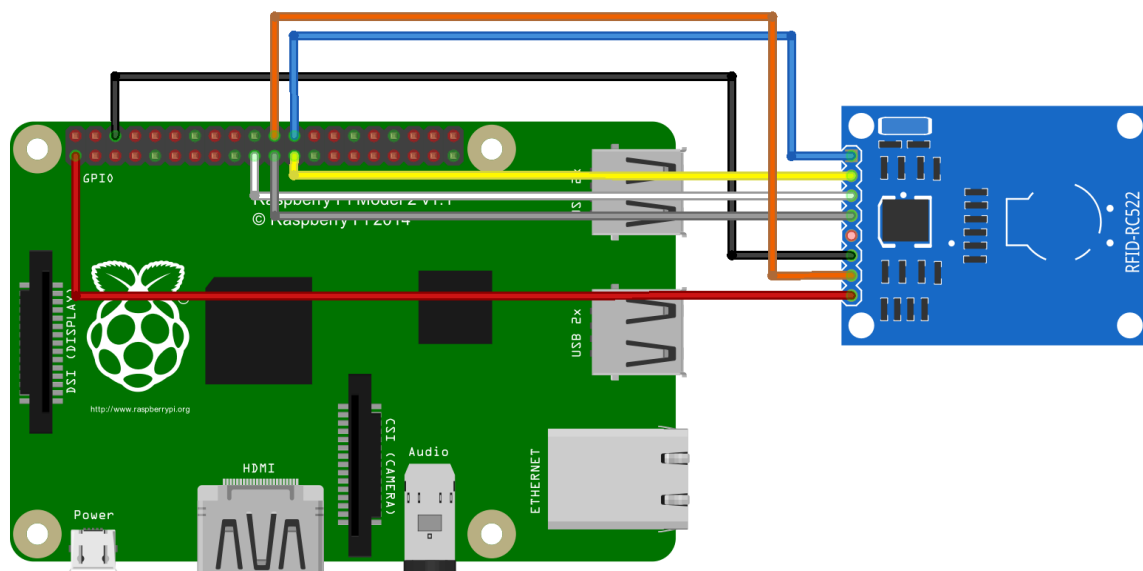


Рисунок 2.4 – Вид соединения Raspberry Pi с RFID

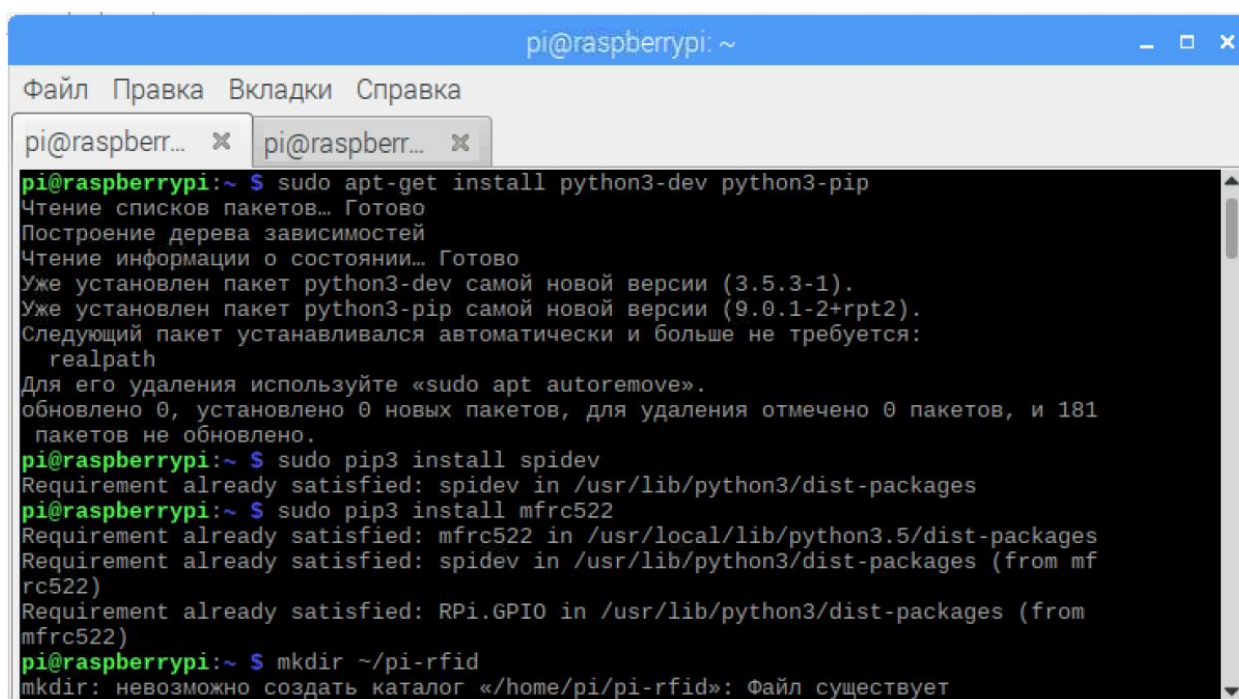
Если видно, что текст «`spi_bcm2835`» появляется в командной строке, то теперь всё готово, чтобы приступить к проверке правильности работы схемы. Как только это будет сделано, можно настроить систему посещаемости на основе RFID [6].

3 Программное средство

3.1 Тестирование RFID RC522

Следующий шаг - установка библиотеки `spidev` на Raspberry Pi, используя команду «`sudo pip3 install spidev`». Библиотека `spidev` предназначена для взаимодействия с интерфейсом RFID-считывателя.

После установки библиотеки `spidev` на Raspberry Pi, нужно приступить к загрузке библиотеки `MFRC522` с помощью команды «`sudo pip3 install mfr522`» (рисунок 3.1). Эта библиотека - то, что будет обрабатывать основную работу для системы посещаемости RFID. Когда и библиотека `MFRC522`, и библиотека `spidev` установлены нужно создать каталог для тестового скрипта командой «`mkdir ~/pi-rfid`» [7].



```
pi@raspberrypi: ~  
Файл Правка Вкладки Справка  
pi@raspberr... x pi@raspberr... x  
pi@raspberrypi:~ $ sudo apt-get install python3-dev python3-pip  
Чтение списков пакетов... Готово  
Построение дерева зависимостей  
Чтение информации о состоянии... Готово  
Уже установлен пакет python3-dev самой новой версии (3.5.3-1).  
Уже установлен пакет python3-pip самой новой версии (9.0.1-2+rpt2).  
Следующий пакет устанавливался автоматически и больше не требуется:  
  realpath  
Для его удаления используйте «sudo apt autoremove».  
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 181  
пакетов не обновлено.  
pi@raspberrypi:~ $ sudo pip3 install spidev  
Requirement already satisfied: spidev in /usr/lib/python3/dist-packages  
pi@raspberrypi:~ $ sudo pip3 install mfr522  
Requirement already satisfied: mfr522 in /usr/local/lib/python3.5/dist-packages  
Requirement already satisfied: spidev in /usr/lib/python3/dist-packages (from mf  
rc522)  
Requirement already satisfied: RPi.GPIO in /usr/lib/python3/dist-packages (from  
mfr522)  
pi@raspberrypi:~ $ mkdir ~/pi-rfid  
mkdir: невозможно создать каталог «/home/pi/pi-rfid»: Файл существует
```

Рисунок 3.1 – Подключение необходимых библиотек

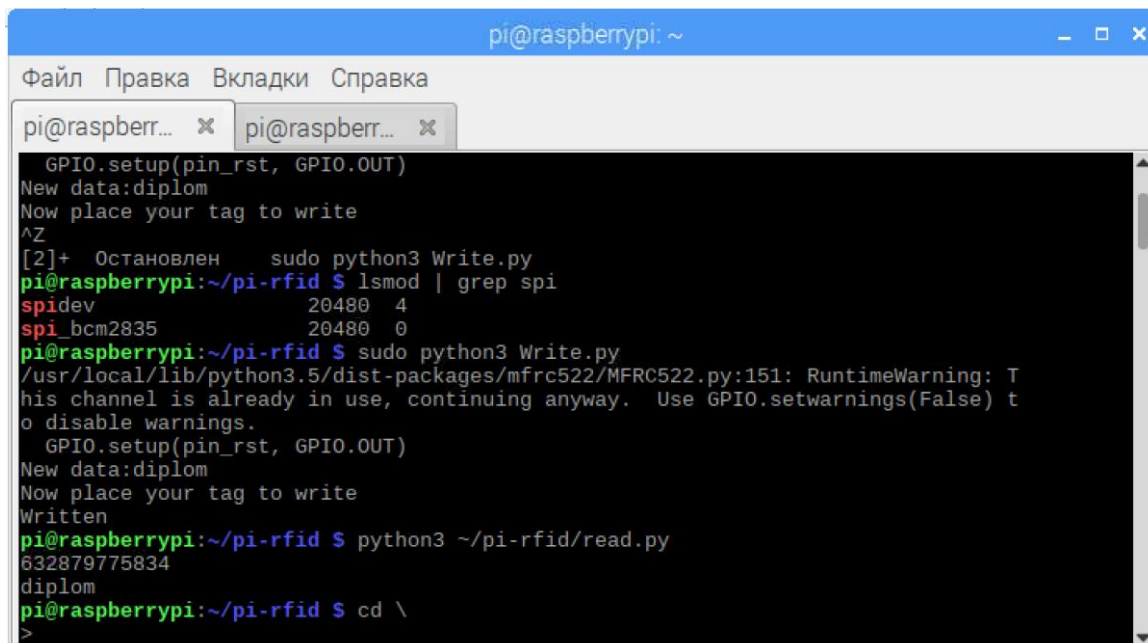
Далее нужно написать короткий скрипт, чтобы проверить, действительно ли RC522 способен считывать карты RFID и что все подключено правильно. Во-первых, следует открыть новый скрипт, выполнив команду «`nano ~/pi-rfid/read.py`» - это создаст файл с именем «`read.py`» в недавно созданном каталоге (рисунок 3.2). Нужно ввести следующие строки (рисунок 3.3):

```
#!/usr/bin/env python  
import RPi.GPIO as GPIO  
from mfr522 import SimpleMFRC522  
reader = SimpleMFRC522()  
try:
```

```

id, text = reader.read()
print(id)
print(text)
finally:
    GPIO.cleanup()

```

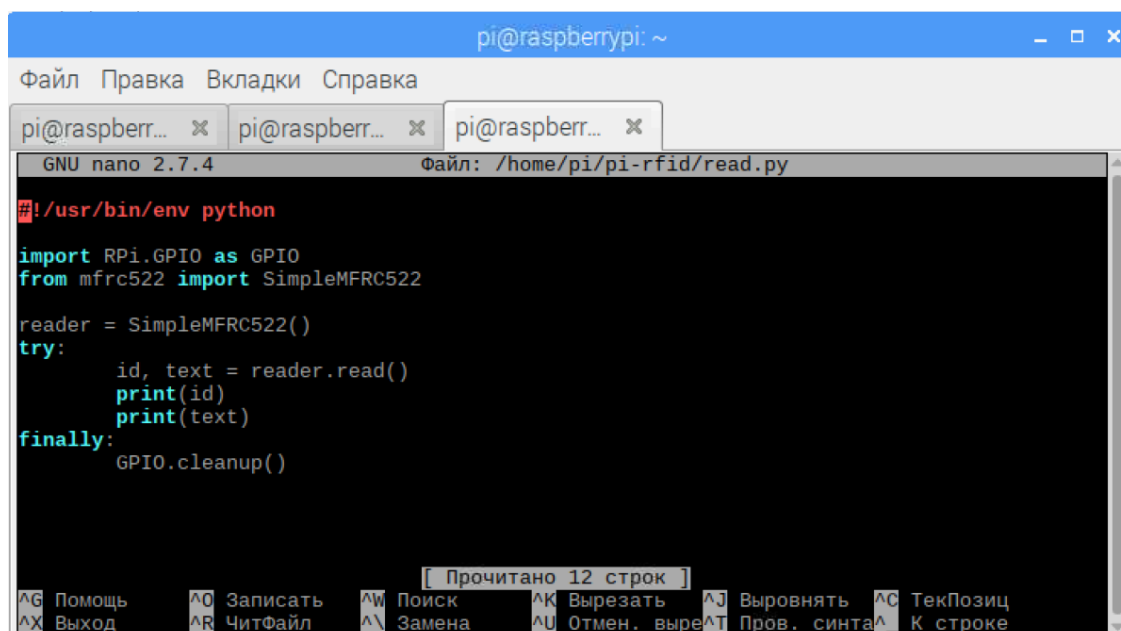


```

pi@raspberrypi: ~
Файл Правка Вкладки Справка
pi@raspberr... x pi@raspberr... x
GPIO.setup(pin_rst, GPIO.OUT)
New data:diplom
Now place your tag to write
^Z
[2]+  Остановлен      sudo python3 Write.py
pi@raspberrypi:~/pi-rfid $ lsmod | grep spi
spidev                20480  4
spi_bcm2835            20480  0
pi@raspberrypi:~/pi-rfid $ sudo python3 Write.py
/usr/local/lib/python3.5/dist-packages/mfrc522/MFRC522.py:151: RuntimeWarning: T
his channel is already in use, continuing anyway. Use GPIO.setwarnings(False) t
o disable warnings.
  GPIO.setup(pin_rst, GPIO.OUT)
New data:diplom
Now place your tag to write
Written
pi@raspberrypi:~/pi-rfid $ python3 ~/pi-rfid/read.py
632879775834
diplom
pi@raspberrypi:~/pi-rfid $ cd \
>

```

Рисунок 3.2 – Создание скрипта для правильной работы устройства



```

GNU nano 2.7.4                                Файл: /home/pi/pi-rfid/read.py
#!/usr/bin/env python

import RPi.GPIO as GPIO
from mfrc522 import SimpleMFRC522

reader = SimpleMFRC522()
try:
    id, text = reader.read()
    print(id)
    print(text)
finally:
    GPIO.cleanup()

[ Прочитано 12 строк ]
^G Помощь      ^O Записать    ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход      ^R ЧитФайл    ^\ Замена     ^U Отмен. выр ^T Пров. синта ^_ К строке

```

Рисунок 3.3 - Редактирование файла «read.py»

После этого сохранить файл, нажав CTRL + X, затем Y и ENTER.

Теперь можно протестировать RFID RC522, запустив скрипт командой «python3 ~/pi-rfid/read.py» и поднести RFID-чип к считывателю.

3.2 Подготовка базы данных системы посещаемости RFID

Прежде чем программировать систему контроля доступа и учета посещаемости, сначала необходимо подготовить и настроить базу данных MySQL. В этой базе данных будет отслеживаться каждая посещаемость RFID-карт и то, кому принадлежит эта RFID-карта. Начать следует с установки MySQL на свой Raspberry Pi, выполнив команду «`sudo apt-get install mysql-server -y`» (рисунок 3.4,3.5).

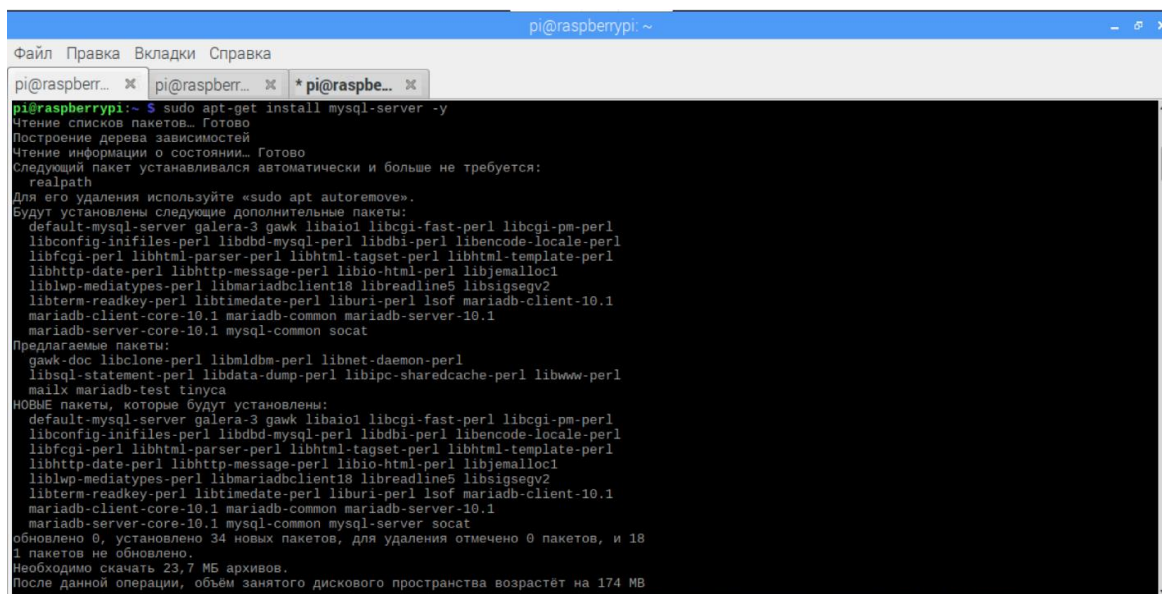


Рисунок 3.4 – Установка MySQL

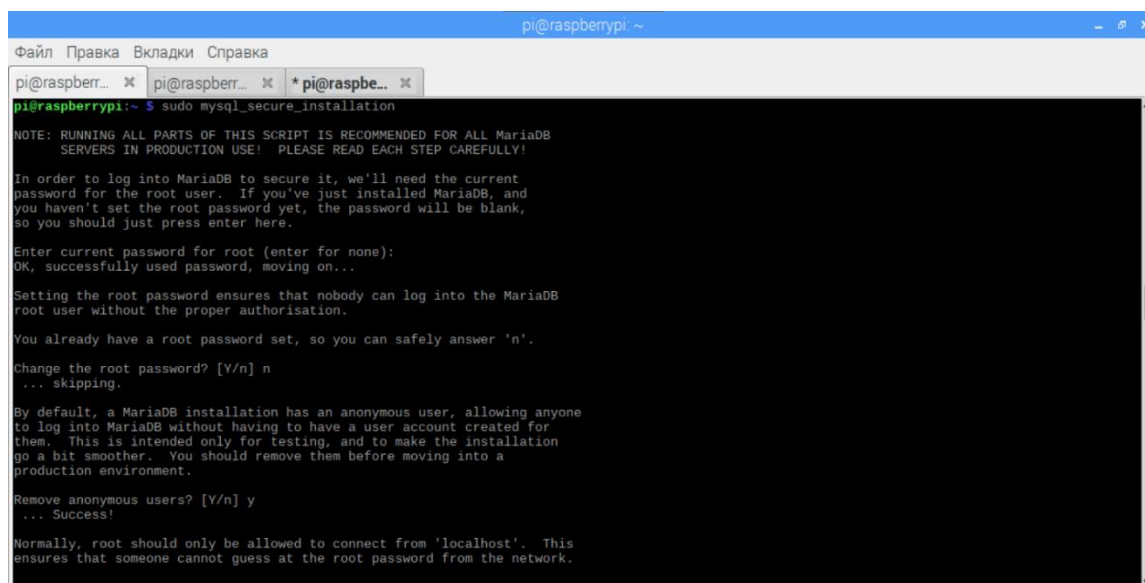


Рисунок 3.5 – Запуск скрипта «безопасной установки»

Затем нужно запустить скрипт «безопасной установки», который поставляется в комплекте с MYSQL.

Этот скрипт проведет некоторые процессы, чтобы сделать сервер MYSQL более безопасным. Нужно запустить этот скрипт, выполнив команду «sudo mysql_secure_installation» в терминале Raspberry Pi.

При появлении запроса требуется убедиться, что установлен новый пароль для корневого сервера MYSQL. Кроме того, нужно ответить «у» на большинство запросов, таких как отключение доступа с правами суперпользователя к серверу MYSQL (рисунок 3.6).

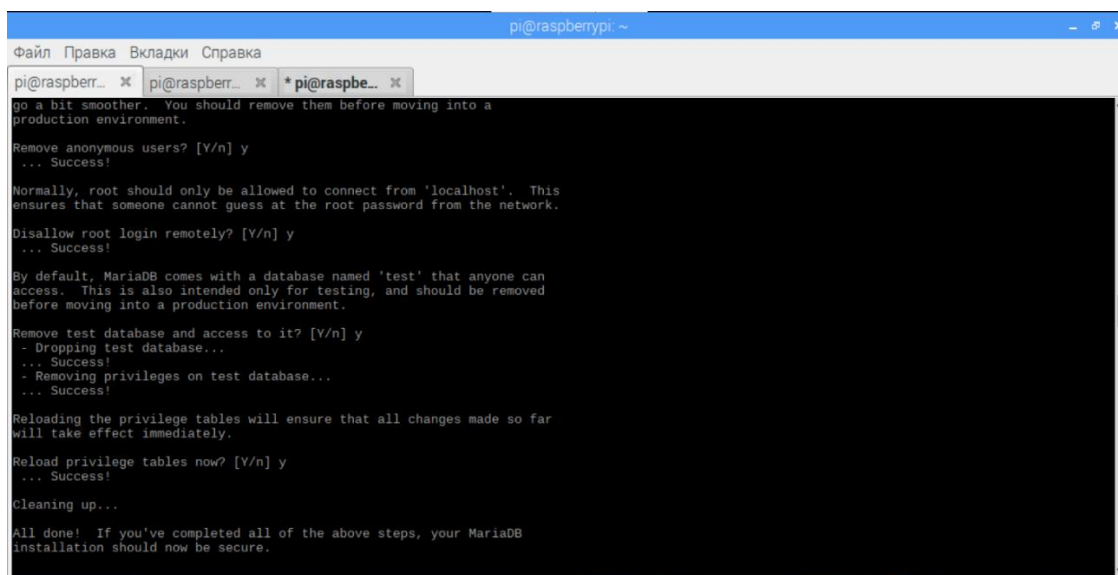


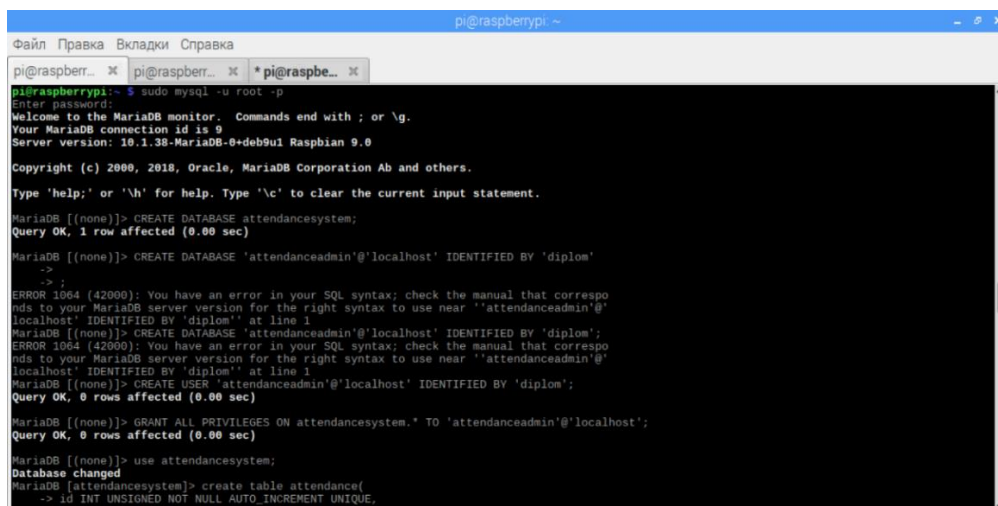
Рисунок 3.6 – Запуск скрипта и установка настроек

Теперь можно загрузиться в инструмент командной строки MYSQL, выполнив команду, далее пользователю будет предложено ввести пароль, который он установил на предыдущем шаге. Поскольку MariaDB при установке использует UNIX_SOCKET в качестве метода аутентификации по умолчанию, пользователь должен войти в систему с помощью суперпользователя, сделать это с помощью «sudo mysql -u root -p» [8].

Далее целесообразно продолжить с создания базы данных, в которой будут храниться все данные, которые используются для системы посещаемости RFID (рисунок 3.7).

Используется команда «CREATE DATABASE attendancesystem;», чтобы создать эту базу данных. После того, как база данных создана, создается пользователь с именем «attendanceadmin», удобно использовать этого пользователя в скриптах Python для чтения из вновь созданной базы данных. Следует убедиться, что для этого установлен пароль, в этом проекте используется «diplom» в качестве пароля. Команда «CREATE USER 'attendanceadmin'@'localhost' IDENTIFIED BY 'diplom';».

После создания пользователя, нужно предоставить ему права доступа к базе данных «посещаемости». Сделать это можно, выполнив «GRANT ALL PRIVILEGES ON attendancesystem.* TO 'attendanceadmin'@'localhost';».



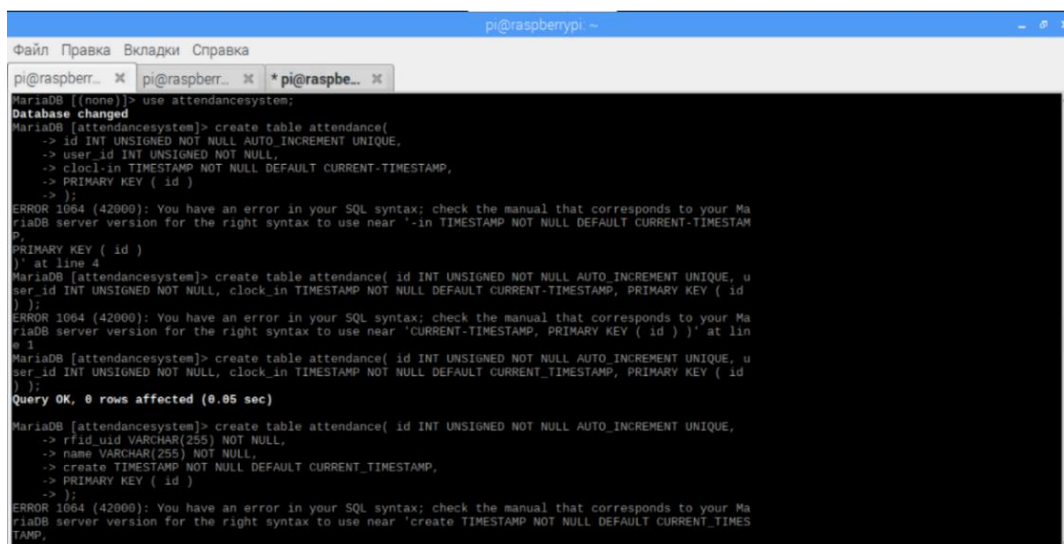
Перед созданием таблиц необходимо использовать команду «use attendancesystem;», чтобы напрямую взаимодействовать с базой данных «посещаемость» (рисунок 3.8). Теперь, имея дело непосредственно с базой данных, которую нужно использовать, можно начать создавать таблицы, в которых будут храниться все данные. Выполнение следующих двух команд создаст таблицы, на которые можно опираться при хранении данных:

Произвести выход из инструмента MYSQL возможно, введя команду «exit» (рисунок 3.9). Для таблицы посещаемости хранятся три фрагмента данных для каждого записанного RFID-сигнала:

а) id - это целое число, которое используется для отслеживания текущей строки и увеличивается автоматически;

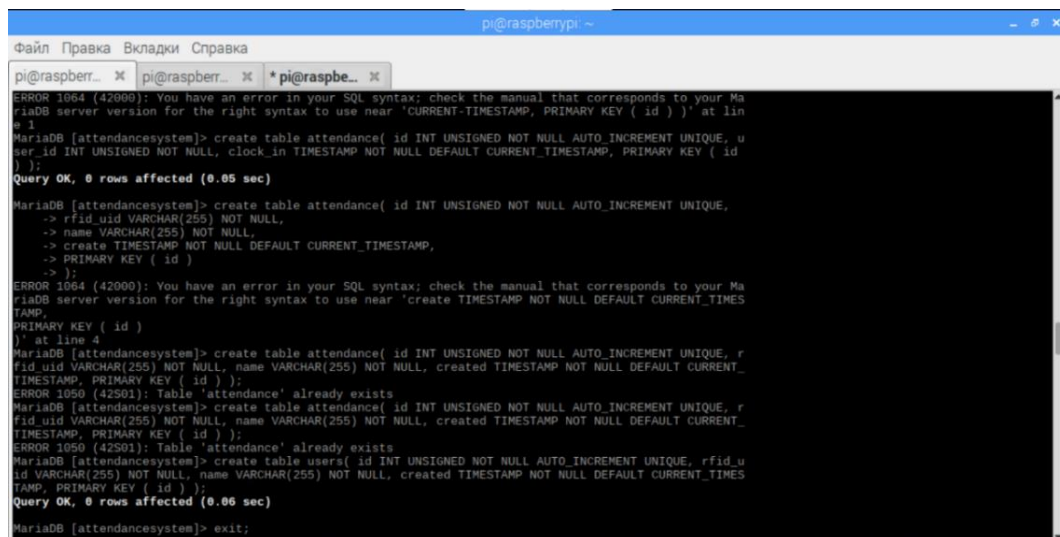
б) user_id - это целое число, используется, чтобы связать посещаемость с пользователем в таблице пользователей с таким же идентификатором;

в) clock_in - эта переменная хранит временную метку SQL. Эта временная метка используется для отслеживания того, когда пользователь подключает свою карту RFID к считывателю RFID.



```
pi@raspberrypi ~  
Файл Правка Вкладки Справка  
pi@raspberrypi:~$ mysql -u root -p  
MariaDB [(none)]> use attendance;  
Database changed  
MariaDB [attendance]> create table attendance(  
-> id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE,  
-> user_id INT UNSIGNED NOT NULL,  
-> clock_in TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
-> PRIMARY KEY ( id )  
-> );  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near '-in TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP' at line 4  
MariaDB [attendance]> create table attendance( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE, u  
ser_id INT UNSIGNED NOT NULL, clock_in TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP, PRIMARY KEY ( id  
));  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'CURRENT_TIMESTAMP, PRIMARY KEY ( id )' at line 1  
MariaDB [attendance]> create table attendance( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE, u  
ser_id INT UNSIGNED NOT NULL, clock_in TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP, PRIMARY KEY ( id  
));  
Query OK, 0 rows affected (0.05 sec)  
MariaDB [attendance]> create table attendance( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE,  
-> rfid_uid VARCHAR(255) NOT NULL,  
-> name VARCHAR(255) NOT NULL,  
-> create TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
-> PRIMARY KEY ( id )  
-> );  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'create TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,' at line 4  
MariaDB [attendance]> exit;
```

Рисунок 3.8 – Внесение изменение в базу данных



```
pi@raspberrypi ~  
Файл Правка Вкладки Справка  
pi@raspberrypi:~$ mysql -u root -p  
MariaDB [(none)]> use attendance;  
Database changed  
MariaDB [attendance]> create table attendance( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE, u  
ser_id INT UNSIGNED NOT NULL, clock_in TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP, PRIMARY KEY ( id  
));  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'CURRENT_TIMESTAMP, PRIMARY KEY ( id )' at line 1  
MariaDB [attendance]> create table attendance( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE, r  
fid_uid VARCHAR(255) NOT NULL, name VARCHAR(255) NOT NULL, created TIMESTAMP NOT NULL DEFAULT CURRENT_  
TIMESTAMP, PRIMARY KEY ( id ));  
ERROR 1059 (42S01): Table 'attendance' already exists  
MariaDB [attendance]> create table attendance( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE, r  
fid_uid VARCHAR(255) NOT NULL, name VARCHAR(255) NOT NULL, created TIMESTAMP NOT NULL DEFAULT CURRENT_  
TIMESTAMP, PRIMARY KEY ( id ));  
ERROR 1059 (42S01): Table 'attendance' already exists  
MariaDB [attendance]> create table users( id INT UNSIGNED NOT NULL AUTO_INCREMENT UNIQUE, rfid_u  
id VARCHAR(255) NOT NULL, name VARCHAR(255) NOT NULL, created TIMESTAMP NOT NULL DEFAULT CURRENT_TIMES  
TAMP, PRIMARY KEY ( id ));  
Query OK, 0 rows affected (0.06 sec)  
MariaDB [attendance]> exit;
```

Рисунок 3.9 – Отчет о внесенных изменениях

Для таблицы пользователей хранятся четыре фрагмента данных для каждого добавляемого пользователя:

а) id - это целое число, которое используется для отслеживания текущего пользователя и увеличивается автоматически;

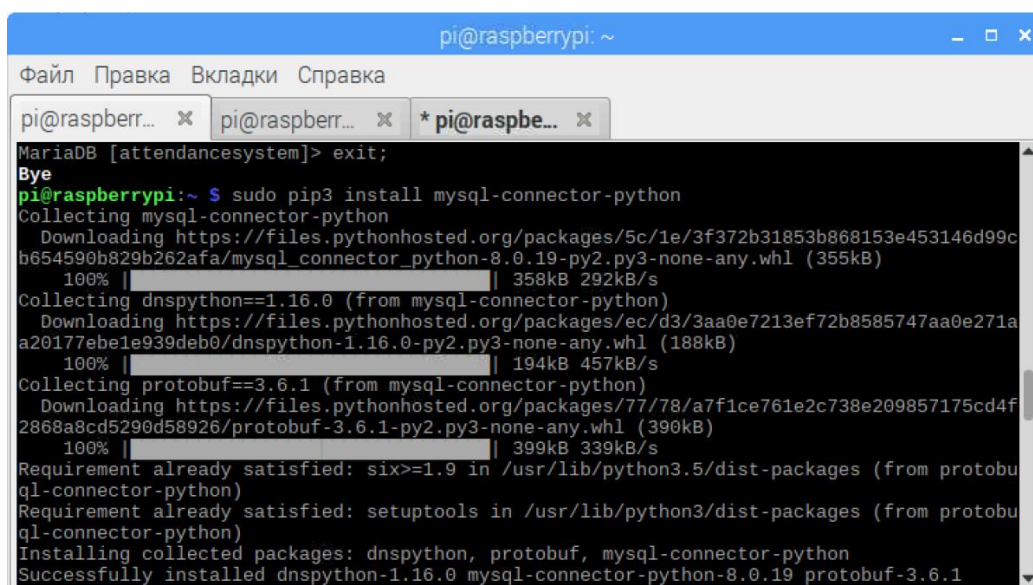
б) `rfid_uid` - эта переменная используется для хранения идентификатора UID, который фиксируется при подведении карты RFID к считывателю;

в) `name` - эта переменная хранит имя человека, которому принадлежит карта RFID;

`created` - используется, чтобы отслеживать, когда пользователь был создан.

3.3 Запись пользователя в системе посещаемости

Прежде чем приступить к написанию скриптов системы посещаемости, нужно сначала установить Python «MySQL Connector» с помощью «`sudo pip3 install mysql-connector-python`» (рисунок 3.10).



```
pi@raspberrypi: ~
Файл Правка Вкладки Справка
pi@raspberr... x pi@raspberr... x *pi@raspb... x
MariaDB [attendancesystem]> exit;
Bye
pi@raspberrypi:~ $ sudo pip3 install mysql-connector-python
Collecting mysql-connector-python
  Downloading https://files.pythonhosted.org/packages/5c/1e/3f372b31853b868153e453146d99c
b654590b829b262afa/mysql_connector_python-8.0.19-py2.py3-none-any.whl (355kB)
  100% |#####| 358kB 292kB/s
Collecting dnspython==1.16.0 (from mysql-connector-python)
  Downloading https://files.pythonhosted.org/packages/ec/d3/3aa0e7213ef72b8585747aa0e271a
a20177ebe1e939deb0/dnspython-1.16.0-py2.py3-none-any.whl (188kB)
  100% |#####| 194kB 457kB/s
Collecting protobuf==3.6.1 (from mysql-connector-python)
  Downloading https://files.pythonhosted.org/packages/77/78/a7f1ce761e2c738e209857175cd4f
2868a8cd5290d58926/protobuf-3.6.1-py2.py3-none-any.whl (390kB)
  100% |#####| 399kB 339kB/s
Requirement already satisfied: six>=1.9 in /usr/lib/python3.5/dist-packages (from protobu
ql-connector-python)
Requirement already satisfied: setuptools in /usr/lib/python3/dist-packages (from protobu
ql-connector-python)
Installing collected packages: dnspython, protobuf, mysql-connector-python
Successfully installed dnspython-1.16.0 mysql-connector-python-8.0.19 protobuf-3.6.1
```

Рисунок 3.10 – Установка Python «MySQL Connector»

Теперь необходимо создать папку, чтобы сохранить все для этого проекта через «`mkdir ~/attendancesystem`».

Пора начать писать первый скрипт на Python для системы контроля доступа и посещаемости. Этот первый скрипт позволит создать пользователя на основе подключенной RFID-карты. После подведения карты RFID скрипт Python попросит пользователя ввести имя, чтобы зарегистрировать эту карту для человека. Написание первой части системы через «`nano ~/attendancesystem/save_user.py`».

Полный код (рисунок 3.11,3.12):

```
#!/usr/bin/env python // чтобы операционная система знала, что этот файл
должен выполняться с использованием Python
import time // импорт библиотеки времени, чтобы перевести скрипт в спящий
режим, чтобы вещи не происходили мгновенно
import RPi.GPIO as GPIO // чтобы запустить функцию очистки после
завершения скрипта
```

```

from mfrc522 import SimpleMFRC522 // используется для упрощения общения
с считывателем RFID
import mysql.connector // чтобы общаться с базой данных, которую была
создана ранее
import Adafruit_CharLCD as LCD
db = mysql.connector.connect(
    host="localhost",
    user="attendanceadmin",
    passwd="diplom",
    database="attendancesystem" // создание соединения с сервером
MySQL. Передача информации, необходимой для установления соединения,
такой как хост, пользователь, имя базы данных и пароль
)
cursor = db.cursor() // создание копии объекта курсора из соединения с базой
данных
reader = SimpleMFRC522() // эта библиотека позволит легко
взаимодействовать с RC522 позже в скрипте, чтобы прочитать ввод от
читателя
lcd = LCD.Adafruit_CharLCD(4, 24, 23, 17, 18, 22, 16, 2, 4);
try:
    while True:
        lcd.clear()
        lcd.message('Place Card to\nregister')
        id, text = reader.read() // этот цикл гарантирует, что приведенный ниже код
будет работать бесконечно, чтобы конечный пользователь мог
зарегистрировать несколько пользователей подряд
        cursor.execute("SELECT id FROM users WHERE rfid_uid="+str(id))
        cursor.fetchone() // чтобы узнать, есть ли у каких-либо строк совпадающий
UID RFID с идентификатором, который получен при чтении карты RFID
        if cursor.rowcount >= 1:
            lcd.clear()
            lcd.message("Overwrite\nexisting user?")
            overwrite = input("Overwrite (Y/N)? ")
            if overwrite[0] == 'Y' or overwrite[0] == 'y':
                lcd.clear()
                lcd.message("Overwriting user.")
                time.sleep(1)
                sql_insert = "UPDATE users SET name = %s WHERE rfid_uid=%s"
            else:
                continue;
        else:
            sql_insert = "INSERT INTO users (name, rfid_uid) VALUES (%s, %s)" //
проверка того, сколько строк было возвращено последним вызовом SQL. Если
вызов SQL возвращает какие-либо строки, нужно указать пользователю, хочет

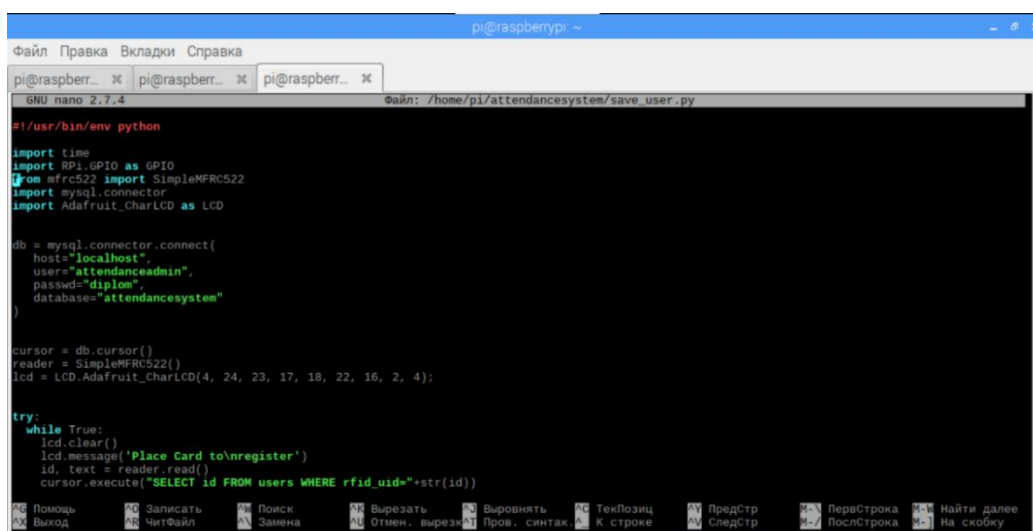
```


ли он перезаписать уже существующего пользователя. Как только функция ввода получила входные данные, идет проверка, равен ли первый символ возвращаемых данных 'Y' или 'y'. Если пользователь отвечает на что-либо, кроме «Y» и «y», на функцию ввода, то происходит возвращение к началу цикла, используя «continue».

```
lcd.clear()
lcd.message('Enter new name')
new_name = input("Name: ")
cursor.execute(sql_insert, (new_name, id))
db.commit()
lcd.clear()
lcd.message("User " + new_name + "\nSaved")
time.sleep(2)
```

finally:

GPIO.cleanup() // как только пользователь ввел имя в консоль и нажал клавишу ввода, происходит переход к использованию объекта курсора для выполнения запроса, который сформирован в предыдущем разделе кода.



```
pi@raspberrypi ~
Файл Правка Вкладки Справка
pi@raspberr... x pi@raspberr... x pi@raspberr... x
GNU nano 2.7.4 Файл: /home/pi/attendancesystem/save_user.py
#!/usr/bin/env python
import time
import RPi.GPIO as GPIO
from mfrc522 import SimpleMFRC522
import mysql.connector
import Adafruit_CharLCD as LCD

db = mysql.connector.connect(
    host="localhost",
    user="attendanceadmin",
    passwd="diplom",
    database="attendancesystem"
)

cursor = db.cursor()
reader = SimpleMFRC522()
lcd = LCD.Adafruit_CharLCD(4, 24, 23, 17, 18, 22, 16, 2, 4);

try:
    while True:
        lcd.clear()
        lcd.message('Place Card to\nregister')
        id, text = reader.read()
        cursor.execute("SELECT id FROM users WHERE rfid_uid="+str(id))
```

Рисунок 3.11 – Код скрипта ~/attendancesystem/save_user.py

Рисунок 3.12 – Продолжение кода скрипта ~/attendancesystem/save_user.py

Запустить скрипт можно, выполнив команду «python3 ~/attendancesystem/save_user.py».

Запуск скрипта показан на рисунке 3.13.

Рисунок 3.13 – Запуск скрипта ~/attendancesystem/save_user.py

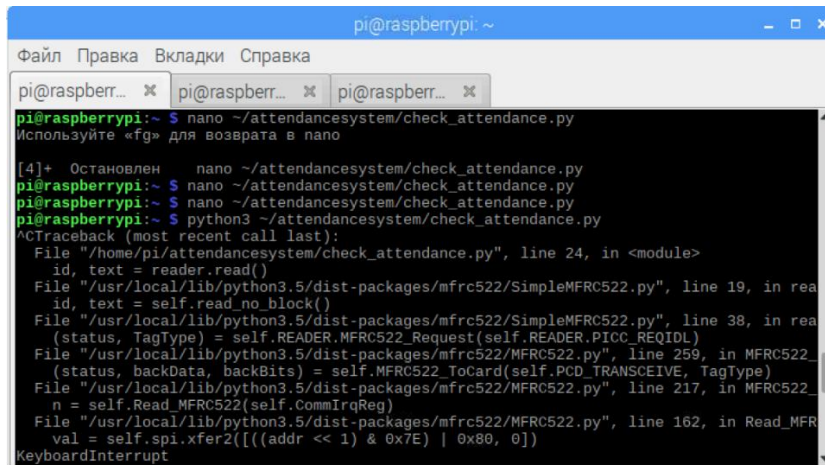
3.4 Запись посещаемости

Когда написан скрипт save_user и он работает правильно, производится переход к скрипту check_attendance. Этот сценарий будет выполняться в бесконечном цикле проверки любых отводов от чипа RFID. Когда кто-то касается своим чипом RFID-считывателя, происходит проверка его идентификатора в базе данных. Если он находит пользователя, устанавливается приветственное сообщение и в таблицу посещаемости вставляется запись с текущей датой и временем.

Начать процесс написания сценария следует с помощью команды «nano ~/attendancesystem/check_attendance.py» (рисунок 3.14).

Полный код (рисунок 3.15,3.16):

```
#!/usr/bin/env python
import time
import RPi.GPIO as GPIO
from mfrc522 import SimpleMFRC522
import mysql.connector
import Adafruit_CharLCD as LCD
db = mysql.connector.connect(
    host="localhost",
    user="attendanceadmin",
    passwd="diplom",
    database="attendancesystem"
)
cursor = db.cursor()
reader = SimpleMFRC522()
lcd = LCD.Adafruit_CharLCD(4, 24, 23, 17, 18, 22, 16, 2, 4);
try:
    while True:
        lcd.clear()
        lcd.message('Place Card to\nrecord attendance')
        id, text = reader.read()
        cursor.execute("Select id, name FROM users WHERE rfid_uid="+str(id))
        result = cursor.fetchone()
        lcd.clear() //выполняется первый бит SQL. Этот оператор SQL извлекает
        «id» и «name» из таблицы «users», где у пользователя тот же RFID ID, что и у
        карты, которая была поднесена к считывателю. Затем берется строка,
        возвращаемая запросом SQL, и ее результат сохраняется
        в переменной «result» для дальнейшего использования
        if cursor.rowcount >= 1:
            lcd.message("Welcome " + result[1])
            cursor.execute("INSERT INTO attendance (user_id) VALUES (%s)",
            (result[0],) )
            db.commit() //сначала проверяется, вернул ли последний запрос SQL какие-
            либо строки. После нужно передать идентификатор пользователя, который
            получен из предыдущего вызова SQL и который сохраняется в результате.
            Наконец, фиксируются изменения в базе данных
        else:
            lcd.message("User does not exist.")
            time.sleep(2)
    finally:
        GPIO.cleanup() // скрипт переводится в спящий режим на две секунды.
        Оператор finally гарантирует, что GPIO очистится после завершения сценария.
```

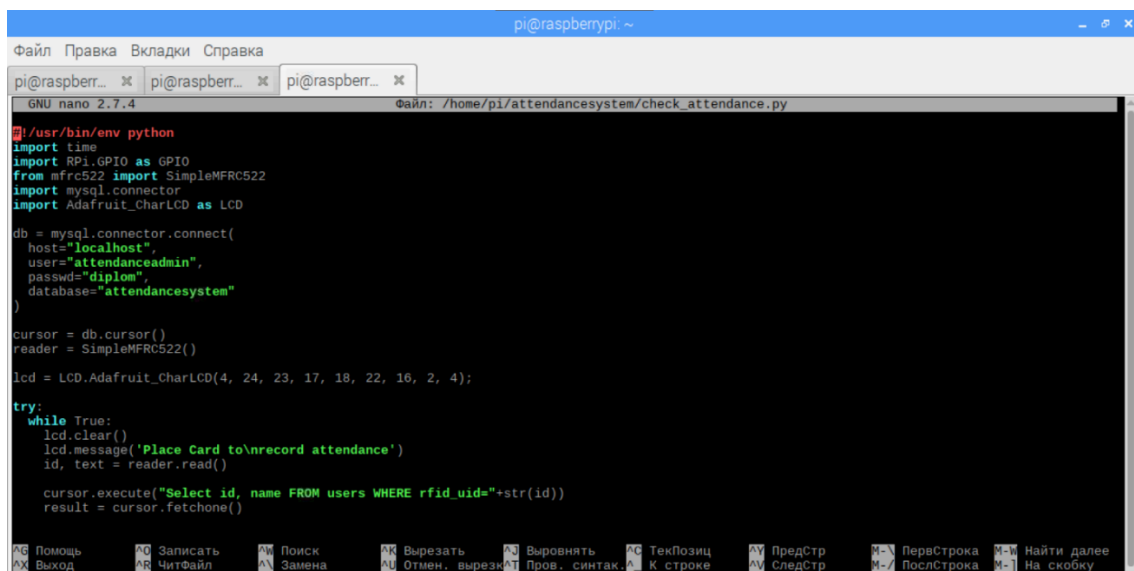


```
pi@raspberrypi ~  
Файл Правка Вкладки Справка  
pi@raspberr... x pi@raspberr... x pi@raspberr... x  
pi@raspberrypi:~$ nano ~/attendancesystem/check_attendance.py  
Используйте «fg» для возврата в nano  
[4]+ Остановлен nano ~/attendancesystem/check_attendance.py  
pi@raspberrypi:~$ nano ~/attendancesystem/check_attendance.py  
pi@raspberrypi:~$ nano ~/attendancesystem/check_attendance.py  
pi@raspberrypi:~$ python3 ~/attendancesystem/check_attendance.py  
^CTraceback (most recent call last):  
  File "/home/pi/attendancesystem/check_attendance.py", line 24, in <module>  
    id, text = reader.read()  
  File "/usr/local/lib/python3.5/dist-packages/mfrc522/SimpleMFRC522.py", line 19, in rea  
    id, text = self.read_no_block()  
  File "/usr/local/lib/python3.5/dist-packages/mfrc522/SimpleMFRC522.py", line 38, in rea  
    (status, TagType) = self.READER.MFRC522_Request(self.READER.PICC_REQIDL)  
  File "/usr/local/lib/python3.5/dist-packages/mfrc522/MFRC522.py", line 259, in MFRC522_  
    (status, backData, backBits) = self.MFRC522_ToCard(self.PCD_TRANSCEIVE, TagType)  
  File "/usr/local/lib/python3.5/dist-packages/mfrc522/MFRC522.py", line 217, in MFRC522_  
    n = self.Read_MFRC522(self.CommIrqReg)  
  File "/usr/local/lib/python3.5/dist-packages/mfrc522/MFRC522.py", line 162, in Read_MFR  
    val = self.spi.xfer2([((addr << 1) & 0x7E) | 0x80, 0])  
KeyboardInterrupt
```

Рисунок 3.14 – Создание скрипта ~/attendancesystem/check_attendance.py

Некоторые блоки кода используются повторно из первого сценария save_user.py, который был написан ранее.

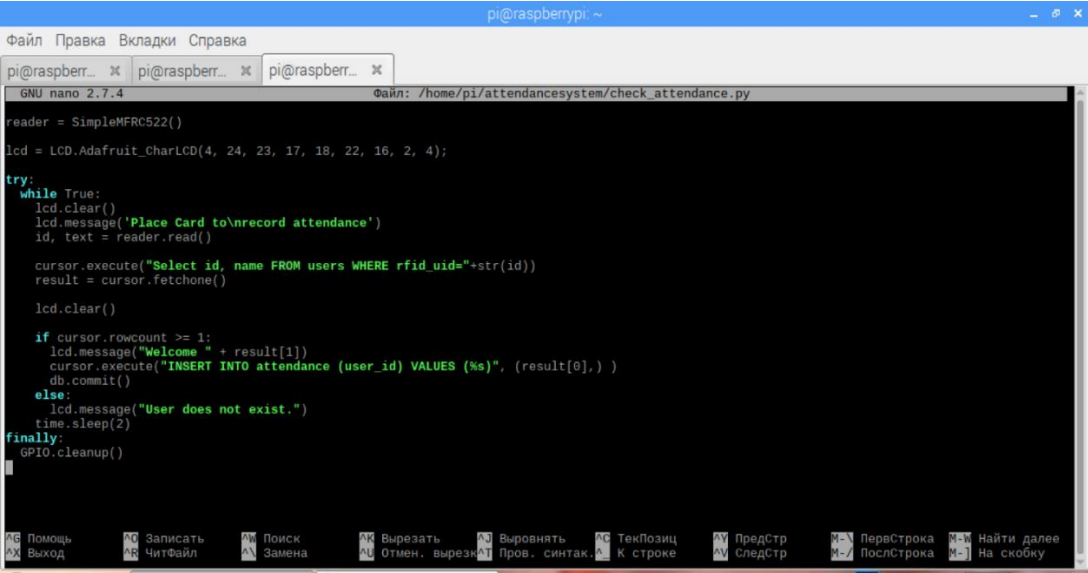
Главное, на что следует обратить внимание при вводе всего кода, - убедиться, что все отступы одинаковы. Два пробела должны разделять каждый уровень.



```
GNU nano 2.7.4 Файл: /home/pi/attendancesystem/check_attendance.py  
#!/usr/bin/env python  
import time  
import RPi.GPIO as GPIO  
from mfrc522 import SimpleMFRC522  
import mysql.connector  
import Adafruit_CharLCD as LCD  
  
db = mysql.connector.connect(  
    host="localhost",  
    user="attendanceadmin",  
    passwd="diplom",  
    database="attendancesystem"  
)  
  
cursor = db.cursor()  
reader = SimpleMFRC522()  
  
lcd = LCD.Adafruit_CharLCD(4, 24, 23, 17, 18, 22, 16, 2, 4);  
  
try:  
    while True:  
        lcd.clear()  
        lcd.message('Place Card to\nrecord attendance')  
        id, text = reader.read()  
  
        cursor.execute("Select id, name FROM users WHERE rfid_uid="+str(id))  
        result = cursor.fetchone()
```

Рисунок 3.15 – Код скрипта ~/attendancesystem/check_attendance.py

Запустив сценарий и введя «python3 ~/attendancesystem/check_attendance.py», можно убедиться в правильности работы разрабатываемой системы.



```
pi@raspberrypi ~  
Файл Правка Вкладки Справка  
pi@raspberr... x pi@raspberr... x pi@raspberr... x  
GNU nano 2.7.4 Файл: /home/pi/attendancesystem/check_attendance.py  
reader = SimpleMFRC522()  
lcd = LCD.Adafruit_CharLCD(4, 24, 23, 17, 18, 22, 16, 2, 4);  
try:  
    while True:  
        lcd.clear()  
        lcd.message('Place Card to\nrecord attendance')  
        id, text = reader.read()  
  
        cursor.execute("Select id, name FROM users WHERE rfid_uid="+str(id))  
        result = cursor.fetchone()  
  
        lcd.clear()  
  
        if cursor.rowcount >= 1:  
            lcd.message("Welcome " + result[1])  
            cursor.execute("INSERT INTO attendance (user_id) VALUES (%)", (result[0],) )  
            db.commit()  
        else:  
            lcd.message("User does not exist.")  
            time.sleep(2)  
finally:  
    GPIO.cleanup()
```

Рисунок 3.16 – Продолжение кода скрипта
~/attendancesystem/check_attendance.py

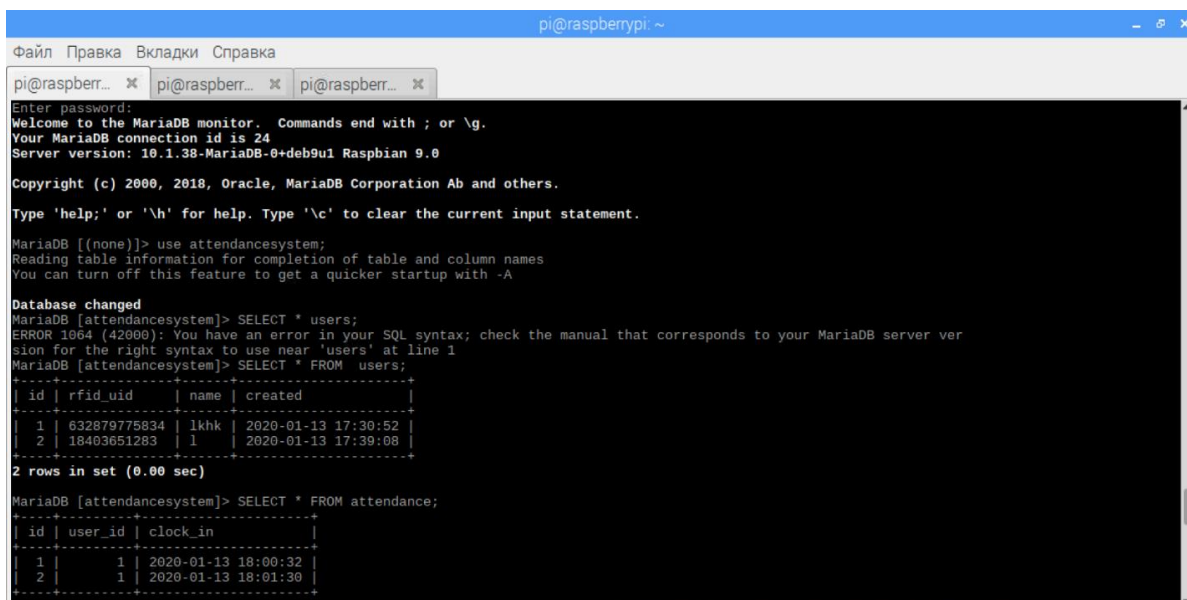
3.5 Проверка базы данных

Далее стоит задача рассмотреть базу данных, чтобы увидеть новые записи. Запустить инструмент командной строки MySQL, выполнив команду «sudo mysql -u root -p». Перед продолжением пользователю будет предложено ввести пароль, который введен для пользователя root. Sudo используется для этой команды, потому что MariaDB по умолчанию использует UNIX_SOCKET для аутентификации.

Как только подключение к командной строке MySQL произведено, нужно использовать команду «use attendancesystem;». Использовать эту команду необходимо, чтобы иметь возможность взаимодействовать с базой данных «система посещений». Напрямую взаимодействуя с базой данных «система посещений», можно начать с проверки всех пользователей, которые были созданы сценарием. Сделать это можно, запустив простой вызов «SELECT * FROM users;», который определяет таблицу «пользователи». Звездочка (*), используемая в запросе, означает, что пользователь хочет охватить все столбцы. Команда требуется, чтобы получить список всех пользователей, доступных в таблице «пользователи» [9].

После завершения проверки таблицы «пользователи» через MySQL, следует рассмотреть таблицу «посещаемости». Как и в предыдущем запросе, просто выбираются все столбцы из таблицы «посещаемости», командой «SELECT * FROM attendance;». Из увиденного можно сослаться на «user_id» обратно на «id» таблицы «users», чтобы увидеть, какой пользователь подключился (рисунок 3.17).

Выйти из инструмента MySQL можно, введя команду «exit;».



```
pi@raspberrypi: ~  
Файл Правка Вкладки Справка  
pi@raspberr... pi@raspberr... pi@raspberr...  
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 24  
Server version: 10.1.38-MariaDB-0+deb9u1 Raspbian 9.0  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
MariaDB [(none)]> use attendancesystem;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
MariaDB [attendancesystem]> SELECT * users;  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server ver  
sion for the right syntax to use near 'users' at line 1  
MariaDB [attendancesystem]> SELECT * FROM users;  
+-----+-----+-----+-----+  
| id | rfid_uid | name | created |  
+-----+-----+-----+-----+  
| 1 | 632879775834 | lkhk | 2020-01-13 17:30:52 |  
| 2 | 18403651283 | 1 | 2020-01-13 17:39:08 |  
+-----+-----+-----+-----+  
2 rows in set (0.00 sec)  
  
MariaDB [attendancesystem]> SELECT * FROM attendance;  
+-----+-----+-----+  
| id | user_id | clock_in |  
+-----+-----+-----+  
| 1 | 1 | 2020-01-13 18:00:32 |  
| 2 | 1 | 2020-01-13 18:01:30 |  
+-----+-----+-----+
```

Рисунок 3.17 – Проверка базы данных

3.6 Создание веб-интерфейса

Этот интерфейс позволяет визуально видеть текущих пользователей и когда они произвели вход в систему. Начало положит создание каталога командой «`sudo mkdir /var/www/html/attendance`», в котором скрипты будут находиться в папке NGINX по умолчанию.

Так как ранее был установлен GIT на Raspberry Pi, поэтому сейчас самое время клонировать скрипты веб-интерфейса в папку, которая была создана ранее. Клонирование кода из репозитория GitHub доступно выполнением команды «`sudo git clone https://github.com/pimylifeup/attendance-system-frontend.git /var/www/html/attendance`». Эта команда будет использовать git для клонирования кода в папку посещаемости. Теперь, когда сценарий клонирован, можно сделать одну модификацию файла «`common.php`», чтобы использовать логины базы данных, которые были установлены ранее. Изменить скрипт с помощью команды «`sudo nano /var/www/html/attendance/common.php`».

Для скриптов интерфейса используется «`medoo`» и «`bootstrap`». Medoo - это легкая платформа для работы с базами данных, подобная той, которая создана для системы посещаемости. Bootstrap, с другой стороны, является интерфейсной средой, которая значительно упрощает разработку чисто выглядящих внешних интерфейсов, не беспокоясь о написании тонны CSS.

Скрипты клонированы в каталог, можно проверить интерфейс, перейдя по IP-адресу используемого Raspberry Pi и добавив «`/attendance`» в конец URL. После перехода на веб-сайт пользователя приветствует веб-страница. Здесь можно проверить своих текущих пользователей, нажав «Пользователи», или просмотреть посещаемость пользователей, нажав кнопку «Посещаемость» (рисунок 3.18-3.21).

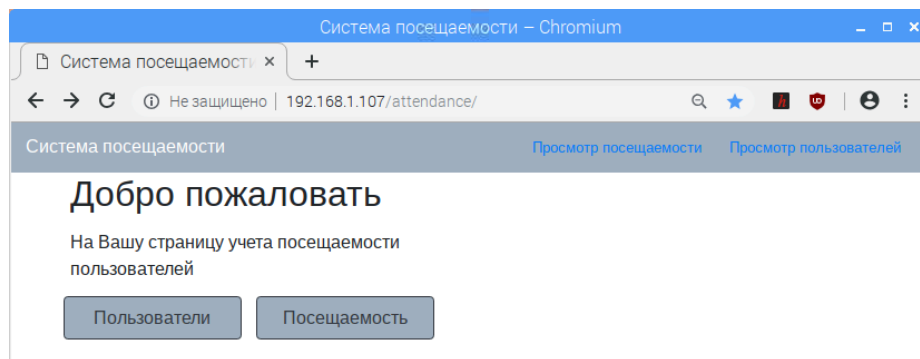


Рисунок 3.18 – Страница приветствия

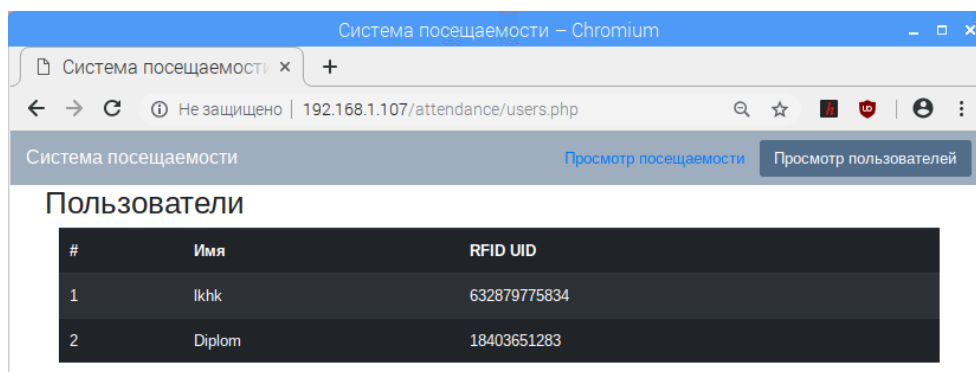


Рисунок 3.19 – Страница пользователей системы посещаемости Система в сборке показана на рисунке 3.22.

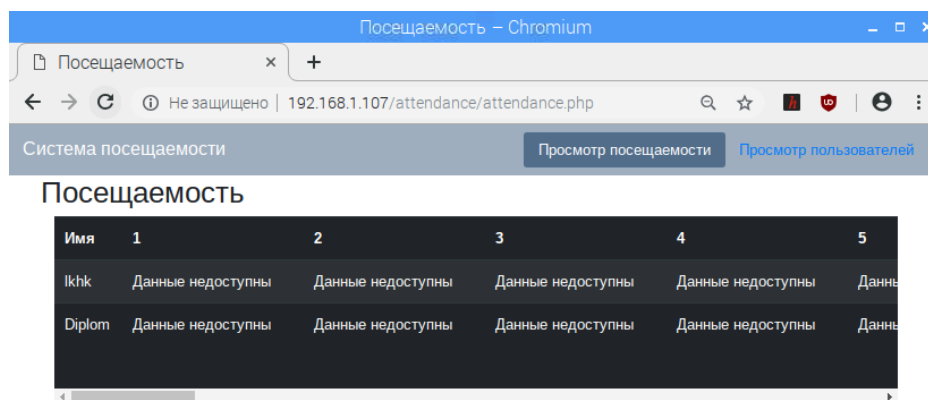


Рисунок 3.20 – Страница посещаемости

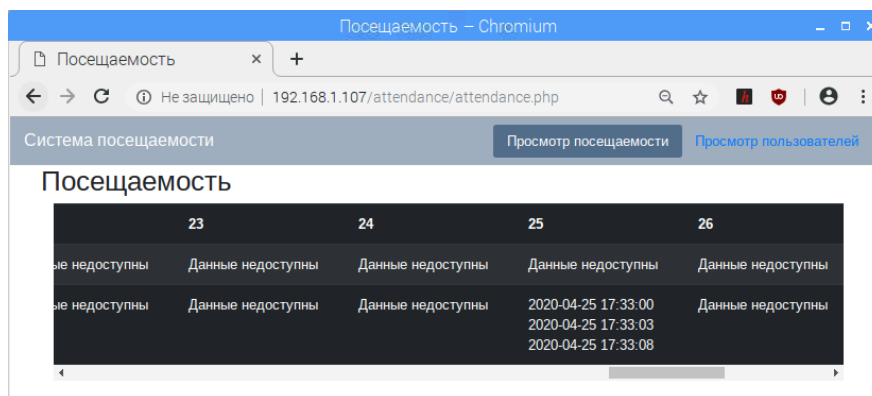


Рисунок 3.21 – Страница посещаемости

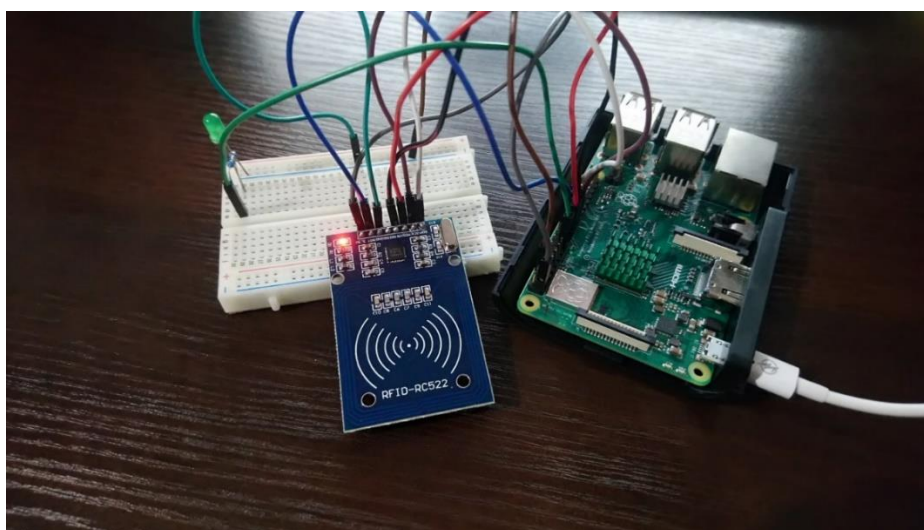


Рисунок 3.22 – Система контроля доступа в сборке

Общая система относительно проста, но охватывает все необходимое для хорошей системы посещаемости.

Возможно дополнительно расширить как бэкэнд, так и внешний интерфейс для реализации новых функций, более приятного пользовательского интерфейса и много другого.

4 Безопасность жизнедеятельности

4.1 Анализ опасных и вредных факторов

4.1.1 Опасные и вредные факторы, воздействующие на персонал в процесс проектирования и эксплуатации объекта

В течение трудовой деятельности работника, находящегося в рабочем пространстве, специализированном помещении, к примеру, офисе, помимо необходимых условий для продуктивности, могут сопутствовать также вредные и опасные факторы. К подобным вредным факторам физической группы можно отнести: отсутствие чистоты, допущение повышенной запыленности, загазованности воздуха в радиусе рабочего пространства, что приводит к поражению органов дыхания и слизистых оболочек организма, острым и хроническим отравлениям; существенно повышенная температура оборудования, значительно высокая или низкая температура воздуха, влажность воздуха и скорость воздуха, которые могут повлечь за собой солнечный или тепловой удары, обморожения, болезни дыхательных путей; превышающий допустимые значения уровень шума, который вызывает снижение и притупление слуха; повышенный уровень вибрации, к неврозу; высокий уровень электромагнитного излучения или иных лучистых энергий, напряженности электрического и магнитного полей, последствиями которых служат болезни органов зрения; нехватка естественного освещения или же его отсутствие, недостаток освещенности рабочей зоны или же напротив повышенная яркость света, сниженная контрастность, блескость от рабочей поверхности, предметов, находящихся вдали или близи, вызывающие раздражение, ухудшение зрения, близорукость.

Факторы химической группы: высокая концентрация токсичных веществ, химикатов и сырья, приводящая к острым и хроническим отравлениям организма, опухолям и ожогам.

Психофизиологическая группа факторов: физические перегрузки динамические и статические, малоподвижность (гиподинамия), которые способствуют появлению грыжи, смещению межпозвоночного диска, расширению вен, образованию тромбов; психические и нервные перегрузки в виде монотонной работы, вызывающие нарушения и заболевания нервной системы.

Опасные факторы включают в себя: технические, то есть недостатки защитных или ограждающих средств, выход из строя машин инструментов, работающих установок и механизмов, обвал конструкций, падения с высоты по причине отсутствия защитных приспособлений, недоработка технологий; организационные: неправильная проектно-техническая документация, допущение к работе людей, не прошедших инструктаж и технику безопасности. Персональные: игнорирование требований безопасности, пренебрежение техникой безопасности, личной безопасностью, использованием средств коллективной и индивидуальной защиты [10].

4.1.2 Микроклимат. Влияние компьютера на рабочего

Сделаем анализ участка, где находятся работники:

- а) размеры офиса: 12х10х4 (соответственно длина, ширина, высота, м);
- б) работа на персональных компьютерах - помещения, где сотрудники работают на персональных компьютерах, должны иметь естественное и искусственное освещение;
- в) основными источниками искусственного освещения являются светильники – 12 шт., в каждом осветительном приборе 2 люминесцентные лампы;
- г) цвет комнаты (белый потолок, светлые стены, темный пол) и находящейся в ней мебели способствуют формированию соответствующих условий для зрительного восприятия;
- д) нормированное значение освещенности для работ средней точности IV разряда, подразряда А составляет 300 лк;
- е) коэффициент естественной освещенности для данного помещения при условии совмещенного освещения составляет 0,5, учитывая характер работ: средней точности;
- ж) для возможного предохранения от блескости окон применяются жалюзи.

Несоответствие освещенности требуемым нормируемым значениям может выступать причиной травм работников на производстве, ухудшения их самочувствия или повышения утомляемости оных, снижения трудовой производительности, снижение качества выпускаемой продукции и так далее. Помещения, где постоянно присутствуют люди должны иметь естественное освещение.

Метеорологические условия (микроклимат) помещений представляет собой климат внутренней среды данных помещений, определяемый действующей на организм человека совокупностью параметров температуры, влажности и скорости движения воздуха, а также теплового излучения.

Гигиенические нормы микроклимата на рабочем месте зависят от категории работы по степени физической нагрузки и от периода года: теплого или холодного. Оценка микроклимата проводилась на основе измерений его параметров на всех местах пребывания работника в течение смены и сопоставления с нормативами СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений».

Гигиенические требования к микроклимату помещений:

- а) температура воздуха;
- б) температура поверхностей;
- в) относительная влажность воздуха;
- г) скорость движения воздуха.

Оптимальные величины показателей микроклимата необходимо соблюдать на рабочих местах помещений, на которых выполняются работы операторского типа, связанные с нервно-эмоциональным напряжением.

Т а б л и ц а 4.1 - Оптимальные нормы микроклимата для офисных и производственных помещений

Период года	Категория работ	Температура воздуха не более	Относительная влажность воздуха, %	Скорость движения воздуха, метров в секунду
Холодный	Легкая 1а	22–24	40–60	0,1
	Легкая 1б	23–21	40–60	0,1
Теплый	Легкая 1а	23–25	40–60	0,1
	Легкая 1б	22–24	40–60	0,1

К категории 1а относятся работы, производимые сидя и не требующие физического напряжения, при которых расход энергозатрат составляет до 139 Вт.

К категории 1б относятся работы, производимые стоя, сидя или связанные с ходьбой и сопровождающиеся некоторым физическим напряжением; расход энергозатрат составляет 140-174 Вт.

При обеспечении допустимого микроклимата на рабочих местах:

а) перепад температуры воздуха по высоте должен быть не более 3 °С;

б) перепад температуры воздуха по горизонтали, а также ее изменения в течение смены не должен превышать при категориях работ:

1) I а и I б - 4 °С;

в) При температуре воздуха на рабочих местах 25 °С и выше максимально — допустимые величины относительной влажности воздуха не должны выходить за пределы:

1) 70 % - при температуре воздуха 25 °С;

2) 65 % - при температуре воздуха 26 °С;

3) 60 % - при температуре воздуха 27 °С;

4) 55 % - при температуре воздуха 28 °С;

г) При температуре воздуха 26-28 °С скорость движения воздуха, для теплого периода времени, должна соответствовать:

1) 0,1 - 0,2 м/с — при категории работ I а;

2) 0,1 - 0,3 м/с — при категории работ I б.

В помещениях, где допустимые нормативные величины показателей микроклимата невозможно установить из-за технологических требований к производственному процессу или экономически обоснованной нецелесообразности, условия микроклимата следует рассматривать как вредные и опасные.

Микроклимат влияет на самочувствие человека, его трудоспособность и протекание физиологических процессов, от которых зависит поддержание постоянства температуры тела. Тепловые воздействия на организм могут явиться причиной быстрого утомления, снижения работоспособности, ослабления сопротивляемости организма к различным заболеваниям: тепловому истощению (слабость, тошнота, головная боль), тепловому удару

(головокружение, возбуждение, дрожь, конвульсии, бред), тепловым судорогам (мышечные спазмы), катаракты глаз [11].

Работа на персональном компьютере. В первую очередь работе за персональным компьютером сопутствует постоянное значительное напряжение органов зрения, причиной которого выступает принцип чтения информации: при обычном чтении текста с бумаги, располагаемой горизонтально на столе, человек воспринимает информацию при падении светового потока на текст с наклоненной головой. При чтении текста с персонального компьютера человек считывает его почти не наклоняя голову, в то же время глаза смотрят прямо вперед (или же почти прямо), источником текста выступает люминесцирующее вещество экрана, поэтому считываемая информация является не отраженным текстом, так как формируется по другую сторону экрана, а самим источником света. Данное явление заставляет функционировать органы зрения длительное время в стрессовом режиме.

Вредное и опасное влияние на органы зрения резко увеличивается при концентрированной работе за персональным компьютером более четырех часов в день. Основными последствиями в случае превышения данного лимита времени выступают: жжение в глазах, покраснение глазных яблок и век, боли в области глазниц, появления чувства наличия инородного тела или песка под веками, помутнение зрения, медленная фокусировка с ближних предметов на дальние. Также длительная работа за монитором, особенно в диалоговом режиме, приводит к перенапряжению нервно-эмоционального характера, расстройству сна, упадку сил и работоспособности, появлению хронической головной боли или даже мигрени, ухудшению общего состояния организма, истощению нервной системы или же появлению депрессии.

Имеют место повышенные статические и динамические нагрузки при работе за персональным компьютером. Они приводят к появлению болей в спине, шее, шейном отделе позвоночника, а также руках.

Симптомами статической нагрузки являются перенапряжения мышц спины, шеи и ног вследствие длительного статического положения при работе за компьютером, а также нерациональная высота рабочей поверхности или сидения, отсутствие наличия опорной спинки, подлокотников, неблагоприятная установка монитора или его положение.

Среди динамических нагрузок основной выступает работа за клавиатурой: при выполнении действий ввода данных возрастает количество движений кистей и пальцев рук, за некоторые периоды данное количество может превышать 60 тыс. Так как всякое нажатие на клавишу связано с сокращением мышц, сухожилия постоянно скользят вдоль костей и соприкасаются с тканями. Данные манипуляции могут повлечь за собой болезненные воспалительные процессы, вследствие наличия систематических нагрузок.

Механизм нарушений, происходящих в организме под влиянием электромагнитных полей, обусловлен их специфическим (нетепловым) и тепловым действием.

Особенное воздействие электромагнитного поля на организм человека определяет биохимические изменения, которые происходят в клетках и тканях. Самыми чувствительными по отношению к подобным воздействиям являются центральная и сердечно-сосудистая системы, не исключены отклонения со стороны эндокринной системы.

С самого начала воздействий может повышаться возбудимость нервной системы, основными симптомами которой являются раздражительность, нарушение сна, эмоциональная неустойчивость. Далее развиваются астенические состояния: физическая и нервно-психическая слабость. Таким образом для хронического воздействия электромагнитного поля характерны: головная боль, утомляемость, ухудшение самочувствия, снижение артериального давления, замедление пульса, боли в сердце.

Тепловое воздействие электромагнитного поля характеризуется повышением температуры тела, локальным нагревом клеток, тканей и органов вследствие перехода электромагнитного поля в тепловую энергию. Интенсивность нагрева зависит от количества поглощенной энергии и скорости оттока тепла от облучаемых участков тела: он затруднен в органах и тканях с плохим кровоснабжением (хрусталик глаза, вследствие чего возможно развитие катаракты). Также негативному воздействию подвергаются печень, поджелудочная железа, мочевой пузырь, желудок, их нагревание может вызвать обострение хронических заболеваний.

Для работы на персональном компьютере установлены специальные требования:

а) площадь одного рабочего места пользователя компьютера с жидкокристаллическим или плазменным монитором должна быть не менее 4,5 кв. м;

б) площадь одного рабочего места пользователя компьютера с монитором на базе электронно-лучевой трубки не менее 6 кв. м;

в) экран видеомонитора должен находиться от глаз пользователя на расстоянии 600–700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов;

г) для предупреждения бликов на экране монитора оконные проемы оборудуются защитными устройствами, не пропускающими дневной свет;

д) рекомендуемая продолжительность непрерывной работы с видеотерминалами и персональными компьютерами составляет не более двух часов;

е) для повышения работоспособности и сохранения здоровья к мерам безопасности относится защита организма с помощью витаминно-минеральных комплексов, которые рекомендуется применять всем, даже практически здоровым пользователям персональных компьютеров;

ж) в помещениях с персональными компьютерами должна ежедневно проводиться влажная уборка;

з) поверхность пола в помещениях эксплуатации компьютеров должна быть ровной, нескользкой, удобной для очистки и уборки;

и) необходимы частые перерывы в работе и эргономические усовершенствования, в том числе оборудование рабочего места так, чтобы исключать неудобные позы и длительные напряжения;

к) помещения с ПК должны оборудоваться системами отопления, кондиционирования воздуха или эффективной приточно-вытяжной вентиляцией.

Напряженность труда – характеристика трудового процесса, отражающая нагрузку преимущественно на центральную нервную систему, органы чувств, эмоциональную сферу работника. К факторам, характеризующим напряженность труда, относятся: интеллектуальные, сенсорные, эмоциональные нагрузки, степень монотонности нагрузок, режим работы. Умственный труд связан с восприятием и переработкой большого количества информации. Умственный труд объединяет работы, связанные с приемом и передачей информации, требующие активизации процессов мышления, внимания, памяти. Умственная работа связана с нервным напряжением, которое зависит от значимости, опасности и ответственности работы. При нервном напряжении возникает тахикардия, рост кровяного давления, увеличение потребления кислорода, длительная умственная нагрузка угнетает психику, ухудшает функции внимания, памяти [12].

4.2 Расчетная часть

4.2.1 Расчет искусственного освещения

В офисном помещении длиной $A = 12$ м, шириной $B = 10$ м, высотой $H = 4$ м требуется создать необходимую для данного типа здания освещенность, используя люминесцентные лампы. Поверхность потолка белая, стены светлые, рабочая поверхность тусклая.

Коэффициенты отражения: потолка $\rho_n = 70$ %, стен $\rho_c = 50$ %, пола $\rho_n = 10$ %. Определим нормированное значение освещенности для выбранного типа работ.

Т а б л и ц а 4.2 - Величина освещенности при искусственном освещении

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд и подразряд зрительной работы		Освещенность, лк, при искусственном освещении
Средней точности	Свыше 0,5 до 1	IV	A	300

Соответственно, нормированное значение требуемой освещенности E равно 300 лк [13].

Выбираем лампы типа ЛБ в светильниках ОД. Так как светильник имеет в поперечной плоскости кривую светораспределения типа Д, то принимаем $\lambda = 1,4$.

Расчетная высота подвеса светильника определяется по формуле:

$$H_p = H - h_c - h_p, \quad (4.1)$$

где H – высота помещения, м;
 h_c – свес светильника, м;
 h_p – высота рабочей поверхности или плоскости, на которой нормируется освещенность, м.

Приняв $h_c = 0$ м, $h_p = 0,75$ м определяем расчетную высоту подвеса светильника H_p :

$$H_p = 4 - 0,75 = 3,25 \text{ м.}$$

Расстояние от светильника до светильника зависит от величины отношения:

$$\lambda = L/H_p, \quad (4.2)$$

где L – расстояние между светильниками или рядами, м;
 H_p – высота подвеса светильника над рабочей поверхностью, м.

Следовательно, определяем оптимальное расстояние между рядами люминесцентных светильников L по формуле:

$$L = \lambda \cdot H_p = 1,4 \cdot 3,25 = 4,55 \text{ м.} \quad (4.3)$$

Рассчитаем количество рядов светильников. Число рядов N при люминесцентных светильниках определяется по формуле:

$$N = B/L = 10/4,55 = 2,19. \quad (4.4)$$

Принимаем $N = 2$.

Зависимость η , коэффициента использования светового потока, от площади помещения, высоты и формы возможно учесть одной комплексной характеристикой - индексом помещения. Индекс помещения i рассчитывается из выражения:

$$i = S / H_p (A+B), \quad (4.5)$$

где S – площадь комнаты, $S = A \cdot B = 12 \cdot 10 = 120 \text{ м}^2$;
 H_p – определенная высота подвеса, $H_p = 3,25$ м;
 A – длина комнаты, $A = 12$ м;
 B – ширина комнаты, $B = 10$ м.

$$i = 120/3,25 (12 + 10) = 1,67.$$

Выбираем коэффициент использования светового потока η .

При коэффициентах отражения $\rho_n = 70 \%$, $\rho_c = 50 \%$, $\rho_p = 10 \%$ и КСС типа Д значение индекса $\eta = 0,72$.

Определяем требуемый поток ламп уже не в светильнике (число светильников пока неизвестно), а в каждом из рядов.

Расчетный поток источников света определяется по формуле:

$$\Phi = E \cdot S \cdot z \cdot k / N \cdot \eta, \quad (4.6)$$

где N - число источников света;

k - коэффициент запаса;

z - коэффициент минимальной освещенности (отношение средней освещенности и минимальной).

Выбираем коэффициент запаса k согласно СНиП 2.04.05. Помещения и территории с нормальными условиями среды (кабинеты и рабочие помещения, жилые комнаты, учебные помещения, лаборатории, читальные залы, залы совещаний, торговые залы и т.д.): коэффициент запаса $k = 1,4$.

Принимаем коэффициент неравномерной освещенности $z = 1,1$, так как в расчетах коэффициент z для светильников с люминесцентными лампами, располагаемых рядами, принимается равным 1,1 [14].

$$\Phi = 300 \cdot 120 \cdot 1,1 \cdot 1,4 / 2 \cdot 0,72 = 55440 / 1,44 = 38500 \text{ лм.}$$

Если в светильники установить по две ЛБ $P_{\text{л}} = 40$ Вт, с потоком 3200 лм, то необходимое число светильников в ряду будет:

$$n = \Phi / \Phi_1, \quad (4.7)$$

где Φ_1 – световой поток одного светильника.

$$N = 38500 / 6400 = 6.$$

Тогда число ламп для всего помещения составит 24. Определяем необходимый световой поток ламп в каждом из рядов:

$$\Phi = 300 \cdot 120 \cdot 1,1 \cdot 1,4 / 24 \cdot 0,72 = 55440 / 17,28 = 3208 \text{ лм.}$$

Делаем проверку выполнения условия:

$$-10\% \leq \frac{\Phi_{\text{л.станд}} - \Phi_{\text{л.расч}}}{\Phi_{\text{л.станд}}} \cdot 100\% \leq +20\% \quad (4.8)$$

Получаем:

$$-10\% \leq -0,25\% \leq +20\%$$

Определяем электрическую мощность осветительной установки [15]:

$$P = N \cdot P_{\text{л}} = 24 \cdot 40 = 960 \text{ Вт.} \quad (4.9)$$

Схема расположения светильников показана на рисунке 4.1.

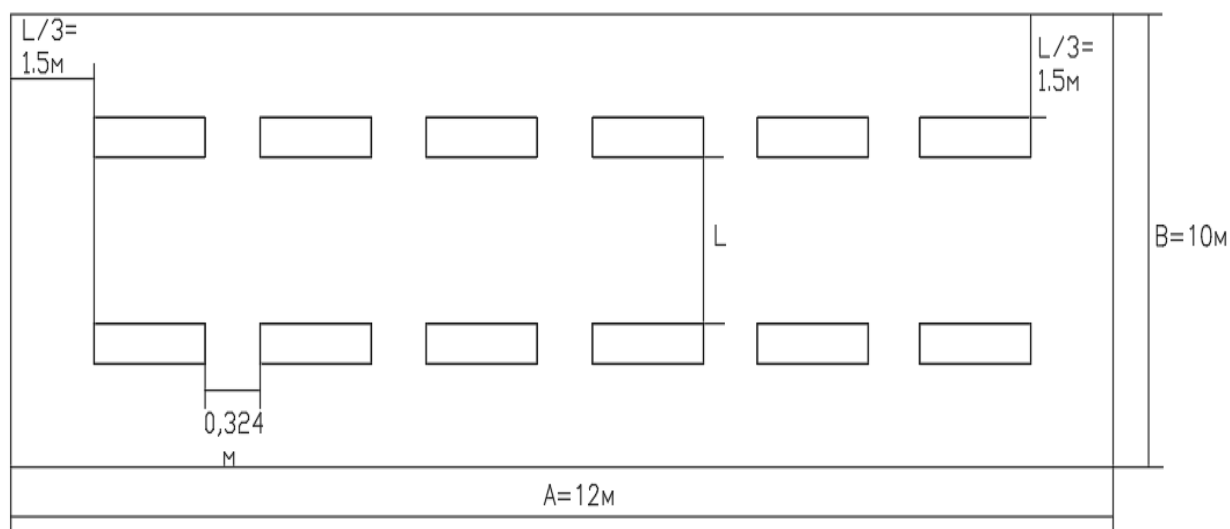


Рисунок 4.1 – Схема расположения светильников

При длине одного светильника 1,23 м их общая длина в ряду составит 7,38 м. Тогда между светильниками в одном ряду могут быть оставлены небольшие разрывы L_p , величину которых несложно рассчитать [16]:

$$L_p = (12 - 7,38 - 3)/5 = 0,324 \text{ м.}$$

4.2.2 Расчет уровня шума, про изводимого персональным компьютером

Высокий уровень шума, который создают печатные устройства, оборудование, предназначенное для кондиционирования воздуха, кулерами, иначе вентиляторами, выступающими в качестве системы охлаждения персональных компьютеров или иных устройств, представляет собой один из небезопасных и неблагоприятных факторов в производственной среде.

Уровень или степень шума, который возникает вследствие функционирования нескольких источников одновременно, следует определять, основываясь на принципе суммирования уровней звукового давления отдельных устройств в выбранной расчетной точке. Определяется уровень шума по формуле:

$$L_i = 10 * \lg \sum_{i=1}^{t=n} 10^{0,1}, \quad (4.10)$$

где L_i - уровень звукового давления i -го источника шума;
 n - количество источников шума.

Вследствие недостижения допустимого значения уровня шума, являющегося результатом расчетов, которые превышают дозвленную норму

давления звука для выбранного рабочего места, принимаются специальные меры. Данные меры направлены непосредственно на снижение уровня шума, они включают в себя: использование звукопоглощающих материалов на поверхности потолка и стен помещения, грамотная планировка используемого аппаратного обеспечения, приглушение шума источника, путем, к примеру, его замены, соответствующая и приемлемая координация рабочего пространства оператора, смена ориентации значимого источника шума [17].

Стандартное рабочее место работника, использующего персональный компьютер в своей трудовой деятельности, оборудовано такими комплектующими и устройствами как: жесткий диск или твердотельный накопитель, помещаемые непосредственно в системный блок, кулеры (вентиляторы), предназначенные для в качестве системы охлаждения для процессора персонального компьютера и аппаратуры внутри системного блока в целом, монитор, компьютерная мышь, клавиатура, сканер и принтер.

Жесткий диск. Обычно применяется в количества одного экземпляра на один системный блок при стандартной работе с малой документацией. При наличии большого объема информации и документов, внедрении расширенных и крупных баз данных может потребоваться и два жестких диска в зависимости от объема (емкости) выбранного устройства. Средняя степень шума жесткого диска варьируется в диапазоне 20-35 дБ, поэтому в качестве точного усредненного значения можно принять 27 дБ. При поиске необходимой существующей информации шум возрастает на 4-6 дБ.

При желании исключения необходимости шумоподавления или во избежание превышения уровня шума целесообразным решением является приобретение твердотельного накопителя. Твердотельный накопитель выступает инновационной улучшенной версией жесткого диска, предоставляя пользователю персонального компьютера бесшумную работу. В данных расчетах будет рассматриваться именно жесткий диск в связи с относительно высокой стоимостью твердотельного накопителя.

Вентиляторы. Идеальная система охлаждения состоит из 3 кулеров для равномерного охлаждения комплектующих персонального компьютера. Однако, в упрощенных системах, не требующих от процессора и аппаратных средств большой нагрузки возможно применение только одного охлаждающего устройства. Уровень шума кулера составляет 26 дБ. При изменениях, производимых в работе системы, шум кулера не изменяется.

Клавиатура. Задействование клавиш для набора текста влечет за собой создаваемый шум, равный 10 дБ.

Компьютерная мышь. Аналогично клавиатуре, при использовании данного периферийного устройства издается характерное клацанье, уровень шума которого в среднем 28 дБ.

Устройства печати. Уровень шума при использовании принтера и сканера соответственно равен 40 дБ и 42 дБ. Применение иного вида принтера, например, струйного, представляет собой увеличение уровня шума на несколько дБ.

Т а б л и ц а 4.3 – Уровни звукового давления и количество источников шума

Источник шума	Количество, штук	Уровень шума, дБ
Жесткий диск	1	27
Кулер	1	26
Компьютерная мышь	1	28
Клавиатура	1	10
Принтер (лазерный)	1	40
Сканер	1	42

Подставив значения уровня шума для каждого вида устройств в формулу, получим:

$$L_i=10\cdot\lg(10^{2,7}+10^{2,6}+10^{3,2}+10^1+10^4+10^{4,2}) = 42,4 \text{ дБ.}$$

При наличии нескольких персональных компьютеров, к примеру, 4, получим:

$$L_4=10\cdot\lg(4\cdot 10^{4,24}) = 48,42 \text{ дБ.}$$

Данное значение получено при учете 4 персональных компьютеров, оснащенных сканером и принтером в том числе. На одно помещение обычно приходится 1-2 подобных периферийных устройства, тогда:

$$L_{пк}=10\cdot\lg(10^{2,7}+10^{2,6}+10^{3,2}+10^1) = 33,9 \text{ дБ.}$$

Значение 33,9 дБ соответствует уровню шума только компьютера, без учета периферийного оборудования. В таком случае наличие 4 персональных компьютера и периферийного оснащения в помещении дают уровень шума:

$$L_{4пк}=10\cdot\lg(4\cdot 10^{3,39}+10^4+10^{4,2}) = 45,52 \text{ дБ.}$$

Значение в 45,52 дБ (и все полученные выше) для рабочего пространства инженеров, операторов, программистов не превышает допустимый уровень шума, равный 50 дБА согласно ГОСТ 12.1.003-83 («Система стандартов безопасности труда. Шум. Общие требования безопасности.»). Кроме того, с учетом малой вероятности использования принтера и сканера, то есть периферийных устройств, в одно и то же время, значение уровня шума будет ниже, также следует учесть, что работа принтера не подразумевает необходимое пребывание оператора в радиусе действия устройства [18].

Вывод: в данной дипломной работе были проанализированы потенциально опасные и вредные факторы, которые в той или иной степени влияют на работника в негативном ключе во время его трудовой деятельности в зоне рабочего пространства. В процессе анализа были выявлены необходимые решения, результатом принятия которых выступит снижение

рисков получения травм и ухудшения состояния здоровья сотрудников. Основопологающим и, следовательно, немаловажным фактором в предотвращении влияния негативных аспектов окружающей среды на рабочем месте является следование рекомендациям в целях личной и коллективной безопасности работников. Необходимой мерой выступает проведение полного инструктажа и ознакомление сотрудников с правилами техники безопасности во избежание создания опасных и рискованных ситуаций, которые могут поставить под угрозу безопасности работников и их коллег.

5 Бизнес-план

5.1 Резюме

Основной целью данного проекта выступает разработка системы контроля доступа и учета посещаемости, суть которой лежит в отслеживании и предоставлении доступа персоналу предприятия или компании через id-карту, которая обладает собственной уникальностью во избежание возникновения ошибок в доступе и рисков безопасности. Возможность ведения учета посещаемости пользователей, имеющих доступ к данной системе, позволяет руководству в любой необходимый момент времени получить всю необходимую информацию об интересующих их пользователях: номер id-карты, имя хозяина карты; точная дата посещения с указанием времени до секунд.

Необходимым для реализации проекта является наличие Raspberry Pi 3 модель B+, micro SD card, периферийные устройства для микрокомпьютера: монитор и клавиатура; RFID-считыватель, id-card или брелок, светодиод и при желании, жидкокристаллический экран для наглядной работы системы при её непосредственной эксплуатации.

Внедрение этой технологии обеспечит компанию возможностью сбора информации о посещениях персонала, мониторинга, установления отсутствия необходимой дисциплины среди подчиненных. Это позволит принимать на работу ответственных, пунктуальных и организованных специалистов, в том числе избежать финансовой нестабильности и нерационального использования денежных средств.

5.2 Описание продукции

Система контроля доступа представляет собой систему, располагаемую на пропускных пунктах предприятий или компаний в целях контроля и ограничения доступа, отслеживания времени, в которое был произведен доступ, а также кем, то есть в целях учета посещаемости. В проекте задействованы механизмы SPI, требуемые для бесконтактного подключения к основополагающему компоненту системы – микрокомпьютеру Raspberry Pi.

5.3 Анализ рынка сбыта

Продукция ориентирована в основном на крупные или малые компании, предприятия, учебные заведения тому подобное, которые составляют около 90% от объема продаж, остальные 10% приходятся на физических лиц, желающих приобрести систему в целях обеспечения безопасности жилого дома или участка.

Основным фактором востребованности данной бизнес-идеи выступает положительная динамика темпов роста. Производство системы контроля доступа характеризуется следующим:

- а) небольшие стартовые вложения;
- б) несложная технология производства;

в) работа с организациями.

Основными преимуществами данной системы выступают:

а) возможность совершенствования технологии;

б) возможность оснащения системы дополнительными функциями без вмешательства в основной функционал.

5.4 Инвестиционный план

В целях осуществления данного проекта необходимо учесть основные разделы, требующие капитальных вложений:

в) комплектующие, выступающие основой сборки устройства системы контроля доступа;

г) фонд оплаты труда работников, разрабатывающих программное обеспечение для системы;

д) накладные расходы.

5.4.1 Расчет капитальных вложений

Инвестиционный капитал (IC – Invest Capital) или капитальные вложения включают в себя стоимость оборудования, монтажных работ и транспортных услуг, т.е. определяется величина капитальных затрат. Для этой цели составляются сметы на приобретение оборудования. Кроме того учитываются затраты на строительство здания, сооружения и т.д.

Т а б л и ц а 5.1 – Стоимость оборудования для реализации проекта

Наименование	Количество штук	Цена за единицу (тг)	Итого (тг)
Raspberry PI 3 b+ (микрокомпьютер, радиаторы, корпус)	1	20480	20480
Дисплей 3.5 LCD	1	6850	6850
Соединительные провода (комплект)	1	765	765
Резистор	1	30	30
Клавиатура Mini Keyboard		4000	4000
Светодиод	1	50	50
Источник питания 5 В – 2.5 А	1	1800	1800
RFID-считыватель RC522	1	560	560
Micro SD карта 16 GB	1	1330	1330
Итого:			35865

Общие капитальные вложения (IC – инвестиционный капитал) [19]:

$$\Sigma IC = K_o + K_c + K_m + K_{tr}, \quad (5.1)$$

где K_o – капитальные вложения на приобретение оборудования (линейных, стационарных и др.);

K_m – капитальные вложения на монтажные работы (5-10% от стоимости оборудования);

K_{tr} – капитальные вложения на транспортные расходы (5-10% от стоимости оборудования);

K_c – капитальные вложения на строительство.

В целях экономии на расходах, цены на комплектующие взяты из доступной виртуальной площадки AliExpress [20].

Расчет велся при закупке товаров в розницу, при развитии производства необходимое оборудование выгоднее закупать оптом, дополнительно снижая расходы. Также следует учесть отсутствие необходимости в большом количестве светодиодов и резисторов, которые обычно продаются комплектом, поэтому цена за единицу товара будет немного выше.

$$K_m = K_o \cdot 0,05 = 1793,25 \text{ тг} \quad (5.2)$$

$$K_{tr} = K_{tr} \cdot 0,05 = 1793,25 \text{ тг} \quad (5.3)$$

Тогда капитальные вложения в проект составят:

$$\Sigma IC = 35865 + 1793,25 + 1793,25 = 39451,5 \text{ тг}$$

5.4.2 Расчет трудоемкости разработки программного обеспечения

Базовый показатель для определения составляющих затрат труда вычисляется по формуле:

$$Q = q \cdot c, \quad (5.4)$$

где Q – условное число команд;

q – коэффициент, учитывающий условное число команд в зависимости от типа задачи;

c – коэффициент, учитывающий новизну и сложность программы.

Выбрать значение коэффициента q можно из таблицы 5.2.

Т а б л и ц а 5.2 – Значения коэффициента q

Тип задачи	Пределы изменений коэффициента
Задачи учета	от 1400 до 1500
Задачи оперативного управления	от 1500 до 1700
Задачи планирования	от 3000 до 3500
Многовариантные задачи	от 4500 до 5000
Комплексные задачи	от 5000 до 5500

Программные продукты по степени новизны может быть отнесены к одной из 4-х групп:

- а) группа А - разработка принципиально новых задач;
- б) группа Б - разработка оригинальных программ;
- в) группа В - разработка программ с использованием типовых решений;
- г) группа Г - разовая типовая задача.

Коэффициент κ определяется из таблицы 5.3, на пересечении групп сложности и степени новизны.

Т а б л и ц а 5.3 – Коэффициенты расчета трудоемкости, κ

Язык программирования	Группа сложности	Степень новизны				Коэффициент κ
		А	Б	В	Г	
Высокого уровня	1	1,38	1,26	1,15	0,69	1,2
	2	1,30	1,19	1,08	0,65	1,35
	3	1,20	1,10	1,00	0,60	1,5
Низкого уровня	1	1,58	1,45	1,32	0,79	1,2
	2	1,49	1,37	1,24	0,74	1,35
	3	1,38	1,26	1,15	0,69	1,5

Программное обеспечение в данном проекте написано на высокоуровневом языке программирования Python, группы сложности 2 при использовании типовых решений в целях выполнения задач оперативного управления. Таким образом коэффициенты $q = 1600$, $\kappa = 1,19$. Тогда базовый показатель для определения составляющих затрат труда будет равен:

$$Q = 1600 \cdot 1,19 = 1904 \text{ команд}$$

Далее необходимо рассчитать время на создание программного продукта. Общее время на создание программы складывается из различных компонентов. Структура общего времени на создание программного продукта представлена в таблице 5.4.

Т а б л и ц а 5.4 – Структура общего времени на создание программного продукта

№ этапа	Обозначение времени данного этапа	Содержание этапа
1	Т _{по}	Подготовка описания задачи
2	Т _о	Описание задачи
3	Т _а	Разработка алгоритма
4	Т _{бс}	Разработка блок-схемы алгоритма
5	Т _н	Написание программы на языке ...
6	Т _п	Набивка программы

7	Тот	Отладка и тестирование программы
8	Тд	Оформление документации, инструкции

Время рассчитывается в человеко-часах, причем Т_{по} берется по фактически отработанному времени, а время остальных этапов определяется расчетным путем, по условному числу команд Q. Определяем время, затраченное на каждый этап создания программного продукта:

1) Т_{по} (время на подготовку описания задачи), берется по факту и составляет (от 3-х до 5-ти дней по 8 часов):

$$T_{по} = 24 \text{ чел/час.}$$

2) Т_о (время на описание задачи) определяется по формуле:

$$T_o = Q \cdot B / (50 \cdot K), \quad (5.5)$$

где В - коэффициент учета изменений задачи, коэффициент В в зависимости от сложности задачи и числа изменений выбирается в интервале от 1,2 до 1,5;

К - коэффициент, учитывающий квалификацию программиста.

Выбрать значение коэффициента К можно из таблицы 5.5.

Т а б л и ц а 5.5 – Коэффициенты квалификации программиста

Опыт работы	Коэффициент квалификации
До двух лет	0,8
2-3 года	1
3-5 лет	1,1 – 1,2
5-7 лет	1,3 – 1,4
более 7 лет	1,5 – 1,6

Коэффициент В, согласно сложности задачи, составляет 1,35, а коэффициент К = 0,8. Следовательно время на описание задачи составит:

$$T_o = 1904 \cdot 1,35 / (50 \cdot 0,8) = 64,26 \text{ чел/час}$$

3) Т_а (время на разработку алгоритма) рассчитываем по формуле:

$$T_a = Q / (50 \cdot K) = 1904 / (50 \cdot 0,8) = 47,6 \text{ чел/час} \quad (5.6)$$

4) Т_{бс} (время на разработку блок-схемы) определяется аналогично Т_а по формуле 3. Соответственно равно 47,6 чел/час.

5) Т_н (время написания программы на языке программирования) определяется по формуле:

$$T_n = Q \cdot 1,5 / (50 \cdot K) \quad (5.7)$$

$$T_H = 1904 \cdot 1,5 / (50 \cdot 0,8) = 71,4 \text{ чел/час}$$

6) T_P (время набивки программы) определяется по формуле:

$$T_P = Q / 50 = 1904 / 50 = 38,08 \text{ чел/час} \quad (5.8)$$

7) $T_{от}$ (время отладки и тестирования программы) определяется по формуле:

$$T_{от} = Q \cdot 4,2/50 \cdot K \quad (5.9)$$

$$T_{от} = 1904 \cdot 4,2/50 \cdot 0,8 = 199,92 \text{ чел/час}$$

8) T_D (время на оформление документации), берется по факту и составляет (от 3-х до 5-ти дней по 8 часов):

$$T_D = 24 \text{ чел / час}$$

Суммарные затраты труда рассчитываются как сумма составных затрат труда по формуле:

$$T = T_{по} + T_o + T_a + T_{бс} + T_H + T_P + T_{от} + T_D \quad (5.10)$$

$$T = 24 + 64,26 + 47,6 + 47,6 + 71,4 + 38,08 + 199,92 + 24 = 516,86 \text{ чел/час}$$

5.4.3 Расчет затрат на разработку программного обеспечения

Затраты на разработку программного обеспечения включают в себя расходы по заработной плате, начислений на заработную плату, амортизацию и прочие расходы и определяются по формуле:

$$C = \text{ФОТ} + \text{ОСН} + A + \text{СЭЭ} + \text{СМиК} + \text{СТО} + \text{СПР} + \text{СНакл}, \quad (5.11)$$

где ФОТ – фонд оплаты труда;

ОСН – отчисления на социальный налог;

A – амортизационные отчисления;

СЭЭ – затраты на электроэнергию;

СМиК – затраты на материалы и комплектующие;

СТО – затраты на техническое обслуживание;

СПР – прочие расходы;

СНакл – накладные расходы.

Заработная плата складывается из двух составляющих: основной заработной платы и дополнительной. Суммарная заработная плата (или фонд оплаты труда, ФОТ) вычисляется как сумма основной и дополнительной заработных плат по формуле:

$$\text{ФОТ} = Z_{осн} + Z_{доп}, \quad (5.12)$$

где Зосн – основная заработная плата, тг;
Здоп – дополнительная заработная плата, тг.

Основная заработная плата рассчитывается по формуле:

$$\text{Зосн} = T \cdot \text{ТС} / (\text{tср} \cdot 8), \quad (5.13)$$

где T – суммарные затраты труда, вычисляемые по формуле;
tср – среднее число дней в месяце, равно 21 дню, умножается на количество часов в рабочем дне – 8;
ТС – тарифная ставка.

Тарифная ставка представляет собой минимальный размер оплаты труда (МРОТ), который с 01.01.2019г в РК составляет 42500 тенге (Закон РК от 30 ноября 2018 года № 197-V «О республиканском бюджете на 2019 – 2021 годы»), увеличенный в зависимости от тарифного коэффициента, соответствующего данному виду работ. Тогда основная заработная плата равна:

$$\text{Зосн} = 516,86 \cdot 42500 / (21 \cdot 8) = 130753,3 \text{ тг}$$

Дополнительная заработная плата составляет 10% от основной заработной платы, рассчитывается по формуле:

$$\text{Здоп} = 0,1 \cdot \text{Зосн} = 0,1 \cdot 130753,3 = 13075,33 \text{ тг} \quad (5.14)$$

Фонд оплаты труда составит:

$$\text{ФОТ} = 130753,3 + 13075,33 = 143828,6 \text{ тг}$$

Социальный налог составляет 9,5 % (ст. 358 п. 1 НК РК) от дохода работника, и рассчитывается по формуле:

$$\text{ОСН} = (\text{ФОТ} - \text{ПО} - \text{ОСМС}) \cdot 9,5\%, \quad (5.15)$$

где ПО – пенсионные отчисления, которые составляют 10% от ФОТ и социальным налогом не облагаются:

$$\text{ПО} = \text{ФОТ} \cdot 10\% = 143828,6 \cdot 0,1 = 14382,86 \text{ тг} \quad (5.16)$$

ОСМС – обязательное социальное медицинское страхование, которое составляет 2% от ФОТ на 2020 г:

$$\text{ОСМС} = \text{ФОТ} \cdot 2\% = 143828,6 \cdot 0,02 = 2876,572 \text{ тг} \quad (5.17)$$

Социальный налог составит:

$$\text{ОСН} = (143828,6 - 14382,86 - 2876,572) \cdot 9,5\% = 12024,07 \text{ тг}$$

Амортизационные отчисления производятся по установленным нормам амортизации, выражаются в процентах к балансовой стоимости оборудования и рассчитываются по формуле:

$$A = \text{Собор} \cdot \text{НА} \cdot N / (100 \cdot 12 \cdot t), \quad (5.18)$$

где НА – норма амортизации;

Собор – первоначальная стоимость оборудования;

N – время использования персонального компьютера;

t – количество рабочих дней в месяце.

Норма амортизации (НА) рассчитывается по формуле:

$$\text{НА} = (\text{Собор} - \text{Сликв} / (\text{Собор} \cdot \text{Тнорм})) \cdot 100\% \quad (5.19)$$

где Сликв – ликвидационная стоимость, составляет 5% от стоимости оборудования;

Тнорм – нормативный срок службы (для микрокомпьютера Raspberry Pi 3 – 3 года).

$$\text{НА} = (35865 - 35865 \cdot 0,05 / (35865 \cdot 3)) \cdot 100\% = 0,3167\%$$

Общее время использования персонального компьютера учитывает лишь время работы на компьютере и рассчитывается по формуле:

$$N = T_a + T_{бс} + T_n + T_p + T_{от} \quad (5.20)$$

$$N = 47,6 + 47,6 + 71,4 + 38,08 + 199,92 = 404,6 \text{ час.}$$

Амортизационные отчисления:

$$A = 35865 \cdot 0,2375 \cdot 404,6 / (100 \cdot 12 \cdot 21) = 182,347 \text{ тг}$$

Затраты на электроэнергию вычисляется по формуле:

$$\text{СЭЭ} = M \cdot k_z \cdot N \cdot \text{СкВт-ч}, \quad (5.21)$$

где M – мощность ЭВМ (0,125 кВт);

k_з – коэффициент загрузки (0.8);

СкВт-ч – стоимость 1 кВт-час электроэнергии (17,81 тг/ кВт·ч);

N - время работы, час.

$$\text{СЭЭ} = 0,125 \cdot 0,8 \cdot 404,6 \cdot 17,81 = 720,5926 \text{ тг}$$

Накладные расходы, связанные с управлением и обслуживанием, содержанием и эксплуатацией оборудования, и прочими дополнительными затратами на обеспечение процессов производства и обращения, составляют 30% от фонда оплаты труда.

Накладные расходы вычисляются по формуле [21]:

$$С_{\text{накл}} = 0,3 \cdot \text{ФОТ} = 0,3 \cdot 143828,6 = 43148,58 \text{ тг} \quad (5.22)$$

Т а б л и ц а 5.6 – Результирующая таблица себестоимости

Статья расходов	Сумма, тенге
Заработная плата, ФОТ	130753,3
	13075,33
Социальный налог, ОСН	12024,07
Амортизационные отчисления, А	182,347
Затраты на электроэнергию, СЭЭ	720,5926
Накладные расходы, С _{накл}	43148,58
Итого:	199904,2

5.4.4 Расчет эксплуатационных расходов

В процессе обслуживания и предоставления услуг связи осуществляется деятельность, требующая расхода ресурсов предприятия. Сумма затрат за год и составит фактическую производственную себестоимость или величину годовых эксплуатационных расходов.

$$\Sigma \text{Э} = \text{ФОТ} + \text{Ос} + \text{М} + \text{Э} + \text{А} + \text{К} + \text{Н}, \quad (5.23)$$

где ФОТ – фонд оплаты (основная и дополнительная заработная плата);

Ос – социальный налог (9,5 % от ФОТ);

М – материальные затраты и запасные части (расходы на запасные части и текущий ремонт составляют 0,5% от капитальных вложений);

Э – расходы на электроэнергию;

А – амортизационные отчисления;

К – кредиты (банковские расходы);

Н – накладные расходы (косвенные расходы, сюда можно отнести все неучтенные расходы – управленческие, хозяйственные, затраты за обучение кадров, транспортные расходы).

Материальные затраты:

$$\text{М} = 0,005 \cdot 39451,5 = 197,2575 \text{ тг}$$

Эксплуатационные расходы составят:

$$\Sigma \text{Э} = 143828,6 + 12024,07 + 197,2575 + 720,5926 + 182,347 + 43148,58 = 200101,4 \text{ тг}$$

В заключении, общая сумма капиталовложений, требуемая для проекта, составляет:

$$\text{К} = \Sigma \text{Э} + \Sigma \text{IC} = 200101,4 + 39451,5 = 239552,9 \text{ тг} \quad (5.24)$$

5.4.5 Расчет доходов, прибыли чистой прибыли и окупаемости

Доходы от реализаций услуг представляют собой выручку от реализации пользователям услуг по установленным тарифам и определяются умножением тарифа на количество реализованных услуг:

$$Д_{\text{реал}} = Ц_{\text{р}} \cdot n, \quad (5.25)$$

где $Ц_{\text{р}}$ – цена реализации с учётом НДС;
 n – количество реализованной продукции, шт.

При определении первоначальной цены изделия задаётся желаемый уровень рентабельности, рекомендуемый в этом случае уровень рентабельности – от 20% до 40% для отрасли телекоммуникации:

$$Ц_{\text{п}} = С \cdot (1 + P / 100), \quad (5.26)$$

где P – рентабельность (20% - 40%);
 $С$ – себестоимость продукции, рассчитанная по формуле 5.4;
 $Ц_{\text{п}}$ – первоначальная цена.

$$Ц_{\text{п}} = 239552,9 \cdot (1 + 30\% / 100) = 240271,6 \text{ тг}$$

Далее определяется цена реализации с учётом НДС:

$$Ц_{\text{р}} = Ц_{\text{п}} + \text{НДС} \quad (5.27)$$

Налог на добавленную стоимость (НДС) 12%, следовательно,

$$Ц_{\text{р}} = Ц_{\text{п}} \cdot 1,12 = 240271,6 \cdot 1,12 = 269104,2 \text{ тг} \quad (5.28)$$

Тогда при реализации системы контроля доступа в количестве одной штуки, доходы от реализации будут равны:

$$Д_{\text{реал}} = 269104,2 \cdot 1 = 269104,2 \text{ тг}$$

Прибыль от реализации услуг исчисляют как разницу между доходами от реализации услуг (или собственными для предприятий где применяется этот показатель) и затратами:

$$П_{\text{реал}} = Д_{\text{реал}} - \sum Э \quad (5.29)$$

$$П_{\text{реал}} = 269104,2 - 200101,4 = 69002,75 \text{ тг}$$

Прибыль облагается корпоративным 20% налогом. После налогообложения прибыль называется чистой прибылью.

$$КПН = 0,2 \cdot 69002,75 = 13800,55 \text{ тг}$$

Чистая прибыль:

$$\text{Пчист} = \text{Преал} - \text{КПН} = 69002,75 - 13800,55 = 55202,4 \text{ тг} \quad (5.30)$$

Расчетный коэффициент экономической эффективности рассчитывается по формуле:

$$E_p = \text{Пчист} / \Sigma \text{IC} = 55202,2 / 39451,5 = 1,39 \quad (5.31)$$

Срок окупаемости рассчитываем по формуле:

$$T = 1/E = 1/1,39 = 0,71 \text{ года} = 8 \text{ месяцев } 16 \text{ дней} \quad (5.32)$$

Вывод: в данной дипломной работе были проведен анализ, в ходе которого был произведен основной расчет необходимых затрат при создании системы контроля доступа, основанном на современных технологиях, для получения максимального результата от проекта при минимуме приведенных затрат. Была рассчитана себестоимость разработанного проекта, его первоначальная цена, чистая прибыль от реализации продукции. Данная система экономически пригодна, так как срок окупаемости составляет меньше года, также система имеет преимущества на рынке в виде возможности усовершенствования и сведения эксплуатационных затрат к минимуму.

Заключение

В ходе выполнения дипломной работы был проведен анализ существующих систем контроля доступа и посещаемости. На основе выполненного анализа была выбрана платформа для реализации собственной системы контроля доступа и посещаемости – Raspberry Pi 3. Так как данный микрокомпьютер на момент выполнения дипломной работы был лидером среди своих конкурентов на рынке электроники, то он был выбран за основу проекта. Были использованы знания и навыки программирования на языке Python и MySQL, а также JavaScript.

Актуальность системы заключается в экономической выгоде внедрения данной системы, возможности её усовершенствования со временем или добавления новых функций, а также её автоматизации, что позволит ускорить работу системы и считывание данных о пользователях.

Данная система может применяться как на крупных предприятиях и производствах, так и в малых компаниях или же даже частными лицами в целях личной безопасности.

Совершенствование системы ограничивается лишь воображением и предоставляет многие возможности по усилению контроля и отслеживанию посещаемости. При наличии бесперебойного питания и доступа к интернету, можно развить мощную систему контроля доступа и учета посещаемости, которая считывает данные каждую секунду. Для подобного использования следует иметь хороший и надежный источник питания и систему охлаждения во избежание допущения перегрева микрокомпьютера.

Список литературы

1. Официальная страница Raspberry Pi – www.raspberrypi.org
2. Петин В.А. Arduino и Raspberry Pi в проектах Internet of Things, 2-е изд. СПб.: БХВ-Петербург, 2019. - 432 с.
3. Макаров С. Л. Arduino Uno и Raspberry Pi 3: от схемотехники к интернету вещей. – Москва.: ДМК Пресс, 2018. - 204 с.
4. Raspberry Pi Cookbook: Software and Hardware Problems and Solutions, 2nd Edition, Simon Monk 14.03.2019.
5. Make: Action: Movement, Light, and Sound with Arduino and Raspberry Pi, Simon Monk, 23.09.2016.
6. Raspberry Pi User guide, Eben Upton, Gareth Halfacree, 2012.
7. Python Programming with Raspberry Pi, 28.04.2017, Sai Yamanoor/
8. Шварц Б., Зайцев П., Ткаченко В. MySQL по максимуму. Санкт-Петербург, 2018. - 864 с.
9. Albert Lukaszewski. MySQL for Python. 2010. - 440 с.
10. И.Ф. Мазалов. Охрана труда. Конспект лекций для студентов специальности 5В073100 – Безопасность жизнедеятельности и защита окружающей среды. – Алматы: АУЭС, 2014. – 90 с.
11. СанПиН РК 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений.
12. СанПиН РК 1.01.004.01 Гигиенические требования к организации и условиям работы с видеодисплейными терминалами и персональными электронно-вычислительными машинами.
13. СНиП РК 2.04.-05.2002 Естественное и искусственное освещение. Государственные нормативы в области архитектуры, градостроительства и строительства.
14. Мананбаева С.Е., Санатова Т.С., Бегимбетова А.С., Бекмуратова Н.С. Безопасность жизнедеятельности. Выполнение раздела дипломных проектов. Учебно-методическое пособие для студентов высших учебных заведений всех специальностей. - Алматы: АУЭС, 2016. - 95с.
15. О.Б. Назаренко. Безопасность жизнедеятельности. Расчёт искусственного освещения. Методические указания к выполнению индивидуальных заданий для студентов дневного и заочного обучения всех направлений и специальностей ТПУ. – Томск: Изд. ТПУ, 2008. – 20 с.
16. А.С. Алданова, Методические указания к выполнению курсовой работы по курсу «Расчет освещения зданий и сооружений» для студентов, обучающихся по специальности 330540 - Светотехника и источники света. – Алматы: АИЭС, 2004. - 35 с.
17. МСН РК 2.04-03-2005 Защита от шума.
18. ГОСТ 12.1.003-83 Система стандартов безопасности труда. Шум. Общие требования безопасности.
19. Г.Ш. Боканова. Методические указания к выполнению экономической части дипломных работ для студентов специальности

5B071900 – Радиотехника, электроника и телекоммуникаций. – Алматы: АУЭС, 2020 – 26 с.

20. Страница виртуальной площадки - www.aliexpress.ru

21. З.Д. Еркешева, Г.Ш. Боканова. Методические указания к выполнению расчетно-графических работ для студентов специальности 5B070400 – «Вычислительная техника и программное обеспечение». – Алматы: АУЭС, 2012. – 36 с.