

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

РАБОТА ПРОВЕРЕНА

Рецензент

_____ Е.А. Зверева
« 05 » _____ июня _____ 2020 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой ЭВМ

_____ Г.И. Радченко
« ____ » _____ 2020 г.

Разработка «умной» кормушки для домашних животных

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ

Руководитель работы,

к.т.н., доцент каф. ЭВМ

_____ Д.В. Топольский
« ____ » _____ 2020 г.

Автор работы,

студент группы КЭ-222

_____ Н.Н. Дубровский
« ____ » _____ 2020 г.

Нормоконтролёр,

ст. преп. каф. ЭВМ

_____ С.В. Сяськов
« ____ » _____ 2020 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

_____ Г.И. Радченко

«___» _____ 2020 г.

ЗАДАНИЕ

на выпускную квалификационную работу магистра
студенту группы КЭ-222
Дубровскому Никите Николаевичу
обучающемуся по направлению
09.03.01 «Информатика и вычислительная техника»

1. **Тема работы:** Разработка «умной» кормушки для домашних животных
утверждена приказом по университету от 24 апреля 2020 г. № 627
2. **Срок сдачи студентом законченной работы:** 1 июня 2020 г.
3. **Исходные данные к работе:**
 - Языки программирования: C++, PHP, SQL, Java.
 - Платформы разработки: Arduino, Android.
 - Библиотеки Arduino: Adafruit PCD8544 Nokia 5110 LCD, NewPing, ESP8266WiFi, ESP8266HTTPClient, EEPROM, Wire, Servo.
 - Библиотеки Android: Volley, Gson, HelloCharts.

- Сервер на операционной системе Debian, с установленными Apache2, MySQL Server/Client, PHP.
- Для отладки базы данных используется PHPMyAdmin.
- REST API для мобильного приложения передает данные в формате Json.
- «Умная» кормушка должна наполнять миску с кормом и водой.
- Датчики должны отслеживать состояние кормушки.
- Должен быть предусмотрен аварийный режим «умной» кормушки.

4. Перечень подлежащих разработке вопросов:

- анализ литературы по теме «умная» кормушка, автоматическая кормушка и концепции «интернет вещей»;
- рассмотрение существующих аналогов «умной» кормушки для домашних животных, оценка их сильных и слабых сторон;
- разработка собственной «умной» кормушки для домашних животных с учетом недостатков аналогов;
- оценка работоспособности разработанной умной кормушки с разными режимами и в разных условиях.

5. Дата выдачи задания: 1 декабря 2019 г.

Руководитель работы _____ /Д.В. Топольский/

Студент _____ /Н.Н. Дубровский /

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	01.03.2020	
Разработка модели, проектирование	01.04.2020	
Реализация прототипа «умной» кормушки для домашних животных	01.05.2020	
Тестирование, отладка, эксперименты	15.05.2020	
Компоновка текста работы и сдача на нормоконтроль	24.05.2020	
Подготовка презентации и доклада	30.05.2020	

Руководитель работы _____ /Д.В. Топольский/

Студент _____ /Н.Н. Дубровский/

АННОТАЦИЯ

Н.Н. Дубровский. Разработка «умной» кормушки для домашних животных. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШЭКН; 2020, 55 с., 28 ил., библиогр. список – 19 наим.

В рамках выпускной квалификационной работы производится детальный анализ литературы по темам: «умная» кормушка, автоматическая кормушка и концепция «Интернет вещей». Организуется разработка программного обеспечения серверной, интерфейсной, микроконтроллерной частей, а так же базы данных. Производится выборка и анализ результатов работы «умной» кормушки, в среде специально разработанных задач. Рассматриваются преимущества и недостатки как в структуре и отдельно взятого компонента «умной» кормушки для домашних животных, так и в отношении аналогов. Доказывается способность системы «умной» кормушки для домашних животных и сервера реализации поставленных задач.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	8
1. ОБЗОР ЛИТЕРАТУРЫ.....	10
1.1. ПОНЯТИЕ И СУЩНОСТЬ «УМНОЙ» КОРМУШКИ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ.....	10
1.2 РАССМОТРЕНИЕ АНАЛОГОВ.....	11
1.3 АНАЛИЗ НЕДОСТАТКОВ АНАЛОГОВ.....	23
2. РАЗРАБОТКА МОДЕЛИ, ПРОЕКТИРОВАНИЕ.....	26
2.1. ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ.....	27
2.2. НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ.....	28
2.3. ВНЕШНИЙ ВИД И СТРУКТУРА «УМНОЙ» КОРМУШКИ.....	29
3. РЕАЛИЗАЦИЯ ПРОТОТИПА «УМНОЙ» КОРМУШКИ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ.....	32
3.1. СТРУКТУРА И ПРИНЦИП РАБОТЫ УСТРОЙСТВА «УМНОЙ» КОРМУШКИ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ.....	32
3.2. СТРУКТУРА И ПРИНЦИП РАБОТЫ СЕРВЕРНОЙ ЧАСТИ.....	38
3.3. МОБИЛЬНОЕ ПРИЛОЖЕНИЕ.....	41
3.4. ПРОТОТИП «УМНОЙ» КОРМУШКИ.....	43
3.5. ОПИСАНИЕ ДАННЫХ.....	44
4. ТЕСТИРОВАНИЕ, ОТЛАДКА, ЭКСПЕРИМЕНТЫ.....	48
4.1. ТЕСТИРОВАНИЕ ДАТЧИКОВ И МОДУЛЕЙ.....	48
4.2. ТЕСТИРОВАНИЕ ПЕРЕДАЧИ ДАННЫХ МЕЖДУ МИКРОКОНТРОЛЛЕРАМИ.....	49
4.3. ТЕСТИРОВАНИЕ АВАРИЙНЫХ СИТУАЦИЙ.....	49
4.4. ТЕСТИРОВАНИЕ ПЕРЕДАЧИ ДАННЫХ НА СВЕРВЕР.....	50
4.5. SMOKE ТЕСТИРОВАНИЕ.....	51

ЗАКЛЮЧЕНИЕ	52
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	53
ПРИЛОЖЕНИЕ А ИСХОДНЫЕ КОДЫ СИСТЕМЫ «УМНАЯ» КОРМУШКА ДЛЯ ДОМАШНИХ ЖИВОТНЫХ	55

ВВЕДЕНИЕ

Актуальность данной темы: на сегодняшний день можно с уверенностью утверждать, что концепция Интернет вещей является надежной сетью устройств, содержащих в себе электронику, программное обеспечение (ПО) и датчики, позволяющие обмен и анализ данных. Концепция имеет высокие темпы развития, и уже глобально вошла в жизнь общества [1]. «Умная» кормушка для домашних животных – одно из явлений концепции Интернет вещей.

Целью представленной выпускной квалификационной работы является разработка «умной» кормушки для домашних животных на основе микроконтроллеров Arduino Mega 2560, Node MCU Amica V3 – ESP 8265, а также сервера Apache с PHP и MySQL.

Для достижения поставленной цели, необходимо решить следующие поставленные задачи:

1. Анализ литературы по теме «умная» кормушка, автоматическая кормушка и концепции «интернет вещей»;
2. Рассмотрение существующих аналогов «умной» кормушки для домашних животных, оценка их сильных и слабых сторон;
3. Разработка собственной «умной» кормушки для домашних животных с учетом недостатков аналогов;
4. Оценка работоспособности разработанной умной кормушки с разными режимами и в разных условиях.

Методы исследования, используемые в выпускной квалификационной работе:

1. Теоретические методы – анализ, сравнение, обобщение, моделирование;
2. Эмпирические методы – наблюдение, эксперимент;

3. Математические методы – обработка количественных данных.

Новизна исследования заключается в том, что «умная» кормушка, представленная в данной работе – основана на концепции «Интернет вещей», является и кормушкой и поилкой одновременно, может вести статистику о количествах засыпания корма в контейнеры, заливания воды, засыпаний корма в миску, активности питомца (когда ел в последний раз или пил), вес питомца, а так же движение питомца. Она имеет несколько режимов работы (обычный, повременной, по приходу питомца). «Умная» кормушка после подключения к беспроводной сети WiFi и выбора режима на сервере не требует больше никаких вмешательств человека или животного (нажатий на кнопку для засыпания корма, ошейников определения питомца поблизости и т.д.). Способна уведомлять о опустошении какого-либо контейнера или разряда батарейки.

1. ОБЗОР ЛИТЕРАТУРЫ

1.1. ПОНЯТИЕ И СУЩНОСТЬ «УМНОЙ» КОРМУШКИ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ

Конкретного определения «Умная кормушка» нет. В данной работе под словосочетанием «Умная кормушка» будет подразумеваться – автоматическая кормушка, работающая по принципам «Интернета вещей». В ее основу входит не только выдача корма по таймеру, а так же анализ полноты запасов корма, поведения животных и подстраивание под их привычки. Помимо этого, «умная кормушка» должна соединяться и работать с «облачными вычислениями» или серверами, иметь контрольную панель на них или отправлять статистику.

Автоматическая кормушка – недавнее изобретение, которое относится к 30–40-м годам XX века. Именно в этот период появился ряд американских патентов на устройства с таймерами [2]. Данные устройства выдавали корм животным только по указанному времени.

Концепция «интернет вещей» – единая сеть, соединяющая окружающие нас объекты реального мира и виртуальные объекты [3].

ИОТ – концепция пространства, в котором все из аналогового и цифрового миров может быть совмещено – это переопределяет отношения людей с объектами, а также свойства и суть самих объектов. Слова Роб Ван Краненбурга. [3].

Облачные вычисления (англ. cloud computing) – технология распределённой обработки данных, в которой компьютерные ресурсы и мощности предоставляются пользователю как Интернет-сервис [4].

1.2 РАССМОТРЕНИЕ АНАЛОГОВ

Для создания особой «умной» кормушки необходимо рассмотрение аналогов, которые могут составить ей конкуренцию. Далее будут рассмотрены характеристики нескольких аналогов «умной» кормушки для домашних животных.

Feed-Ex для сухого корма.

Характеристики:

1. Материал – прочный пластик, не скрипит, не гнется.
2. Позволяет настроить пищевые пропорции.
3. Предотвращает быстрое употребление пищи с возможностью медленной подачи.
4. Можно настроить голосовое обращение.
5. Имеет большой резервуар для корма.
6. Функции программирования сложны для новичков [5].
7. Объем каждого кормления – от 25 гр. до 250гр сухого корма.
8. Объем емкости для корма – 4 л (2 кг сухого корма).
9. Габариты : 22 x 27 x 32см (со вставленным лотком).
- 10.Цвет – желтый.
- 11.Вес (с упаковкой) - 2 кг.
- 12.Комплектация – автокормушка в собранном виде, наружный лоток для корма, инструкция. Батарейки в комплект не входят [6].



Рисунок 1 – Кормушка Feed-Ex для сухого корма.

Структура и принцип работы автоматизированной Feed-Ex кормушке представлены на рисунках 2 и 3.

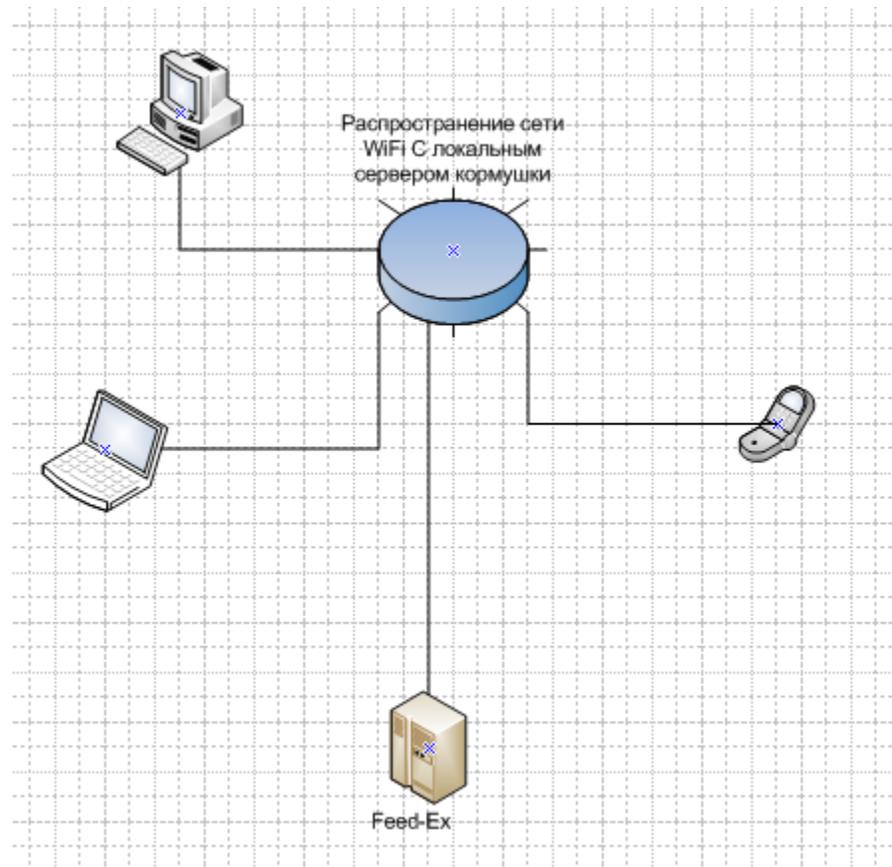


Рисунок 2 – Кормушка Feed-Ex в макро структуре.

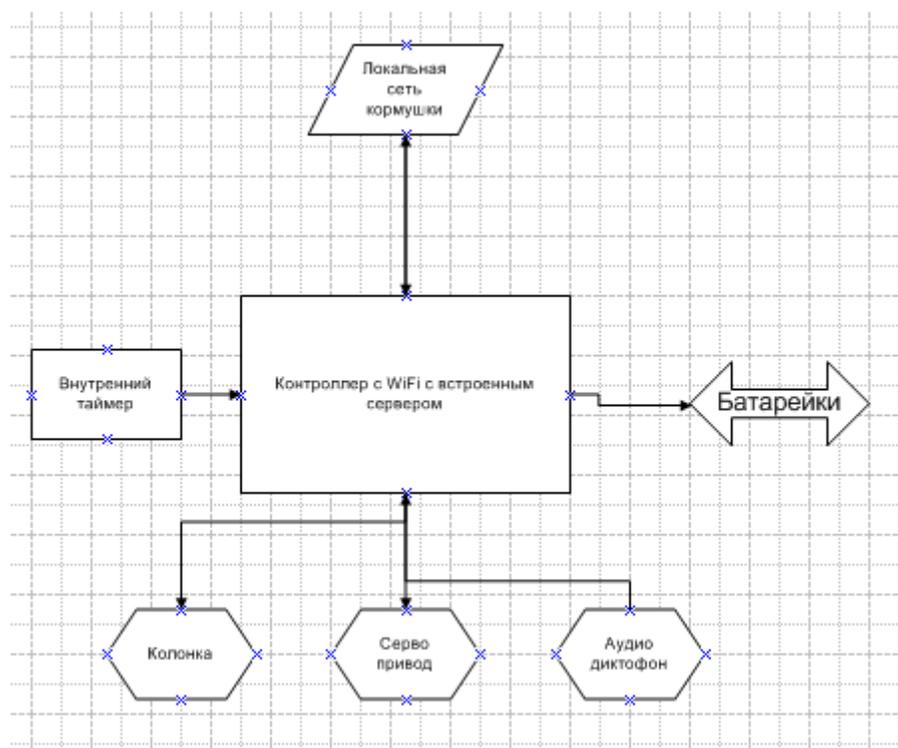


Рисунок 3 – Кормушка Feed-Ex в микро структуре.

Из представленных структур видно, что автоматическая кормушка работает исключительно по таймеру, имеет внутренний WiFi и веб-сервер для настроек и работает в локальной сети своего WiFi, работает от пальчиковых батареек [6].

Автоматическая кошачья кормушка для влажной еды - Feed and Go Automatic Feeder.

Характеристики:

1. Устройство позволяет пользователям дистанционно программировать время подачи.
2. Графики могут быть изменены с использованием Интернета.
3. Довольно прочный корпус.
4. Один вариант подачи запускается одним щелчком мыши.

5. Работает на сухой и влажной пище.
6. Пользователь может запрограммировать 6 сеансов приема пищи с интервалом в несколько минут или часов.
7. Может записывать видео и аудио.
8. Можно настроить голосовое обращение.
9. Поддержка интерфейса Wi-Fi.
10. Встроенные микрофон и WEB-камера.
11. Вмещает 6 порций еды.
12. Крайне проста в управлении.
13. Синхронизация со смартфоном [7].



Рисунок 4 – Кормушка Feed and Go Automatic Feeder для влажного корма.

Структура и принцип работы автоматизированной кормушки Feed and Go Automatic Feeder кормушке представлены на рисунках 5 и 6.

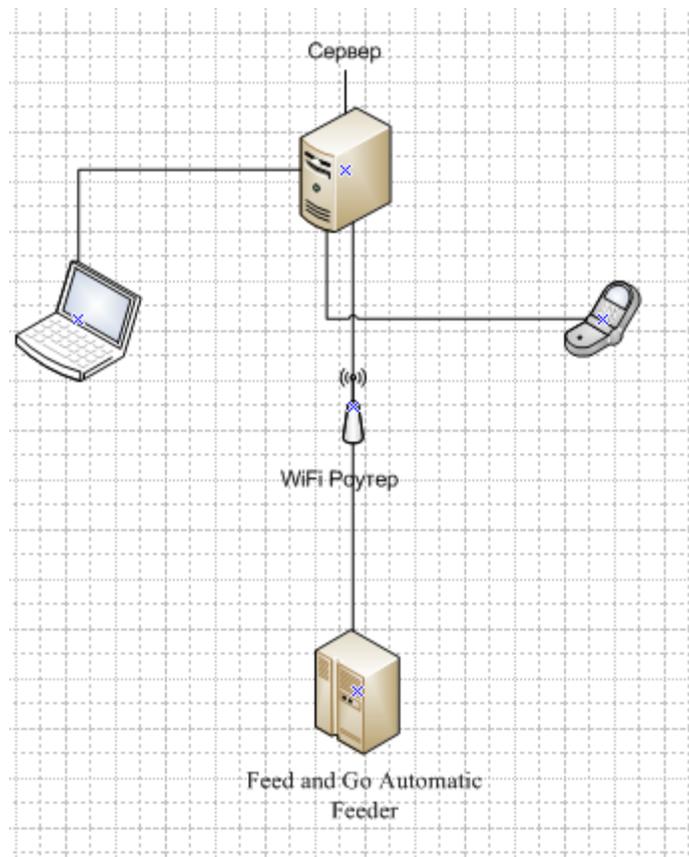


Рисунок 5 – Кормушка Feed and Go Automatic Feeder в макро структуре.

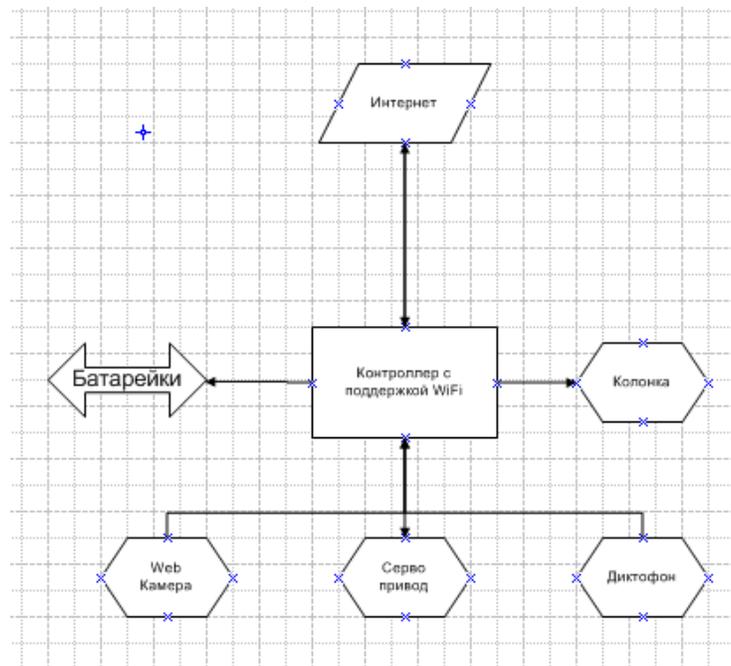


Рисунок 6 – Кормушка Feed and Go Automatic Feeder в микро структуре.

Из представленных структур видно, что автоматическая кормушка работает исключительно по таймеру, имеет способность подключаться к серверу по WiFi, может отправлять видео данные на сервер, работает от пальчиковых батареек [7].

Кормушка SuperFeed CSF-3.

Характеристики:

1. Может настраивать графики подачи.
2. Перерывы питания не требуют перепрограммирования.
3. Работает лучше для мелких гранул.
4. Прочная крышка желоба не позволяет кошкам красть свою пищу.
5. Работает для двух кошек с 8-мя программами кормления [5].



Рисунок 7 – Кормушка SuperFeed CSF-3.

Структура и принцип работы автоматизированной кормушки SuperFeed CSF-3 представлены на рисунках 8 и 9.

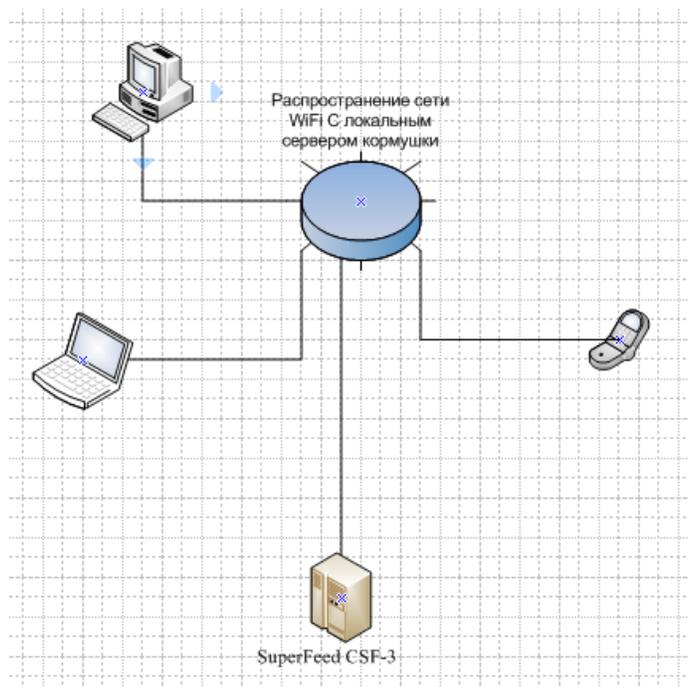


Рисунок 8 – Кормушка SuperFeed CSF-3 в макро структуре.

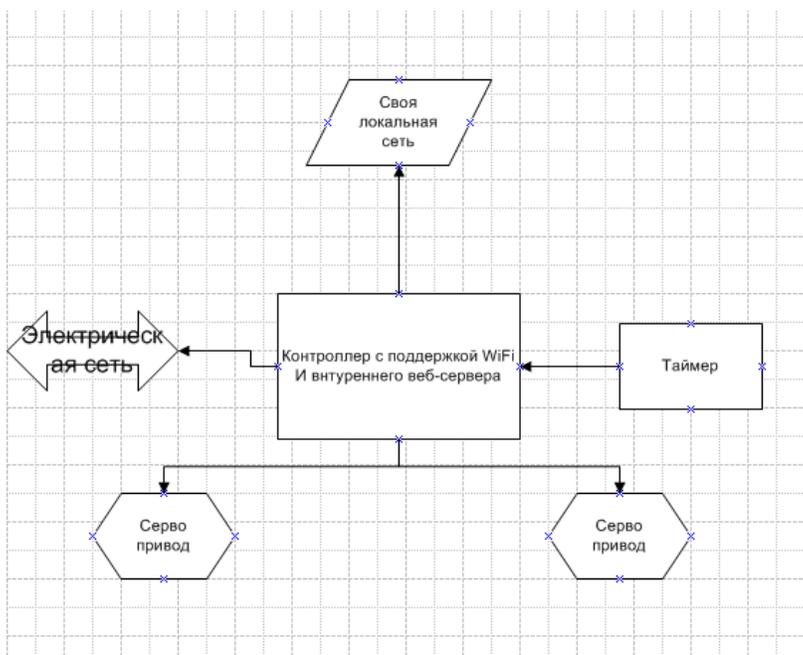


Рисунок 9 – Кормушка SuperFeed CSF-3 в микро структуре.

Из представленных структур видно, что данная кормушка работает исключительно по таймеру, хоть и в разных режимах, имеет внутреннюю WiFi сеть и роутер, работает в локальной сети, работает от сети [5].

LUSMO Автоматический раздатчик еды для контроля веса.

Характеристики:

1. Кормушка позволяет владельцам домашних животных настраивать порции еды.
2. Время приема пищи может быть изменено в разное время.
3. Полный запас может поддерживаться до 10 дней
4. Запираемая крышка
5. Легко читаемый ЖК-монитор для времени и состояния батареи.
6. Устройство не работает с каждым типом корма, особенно с кубическими и длинными [5].



Рисунок 10 – Кормушка LUSMO.

Структура и принцип работы кормушки LUSMO кормушке представлены на рисунке 11. Данную кормушку на макро уровне нет смысла рассматривать, так как она не работает с сетью совсем.

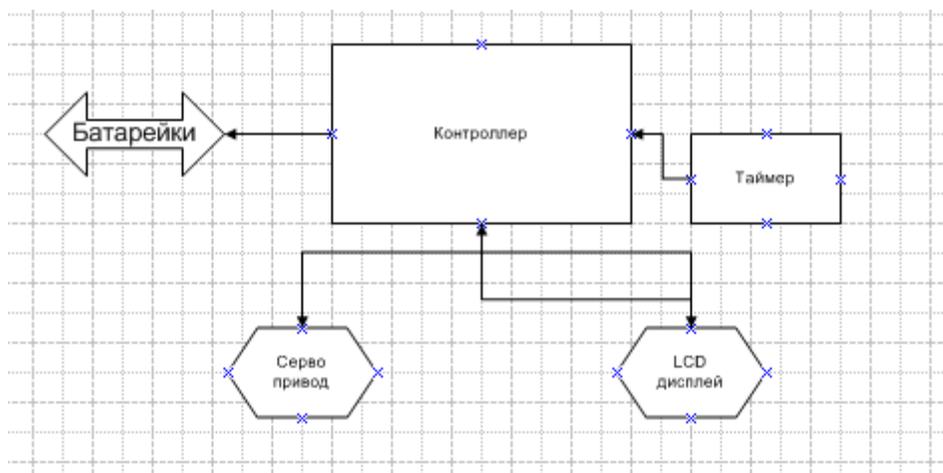


Рисунок 11 – Кормушка LUSMO в микро структуре.

Из представленной структуры видно, что кормушка LUSMO работает только на батарейках, не имеет выхода в сеть [5].

Кормушка Furrytail Smart Feeder.

Характеристики:

1. Материал – ABS пластик.
2. Отверстие для выдачи корма имеет подсветку.
3. Корм подается при помощи спирали.
4. За отслеживание количества корма отвечает пара датчиков, расположенных друг напротив друга на стенках кормушки.
5. Управление производится с мобильного приложения.
6. Интерфейс приложения полностью на китайском языке [8].



Рисунок 12 – Кормушка Furrytail Smart Feeder.

Структура и принцип работы кормушки Furrytail Smart Feeder представлены на рисунках 13 и 14.

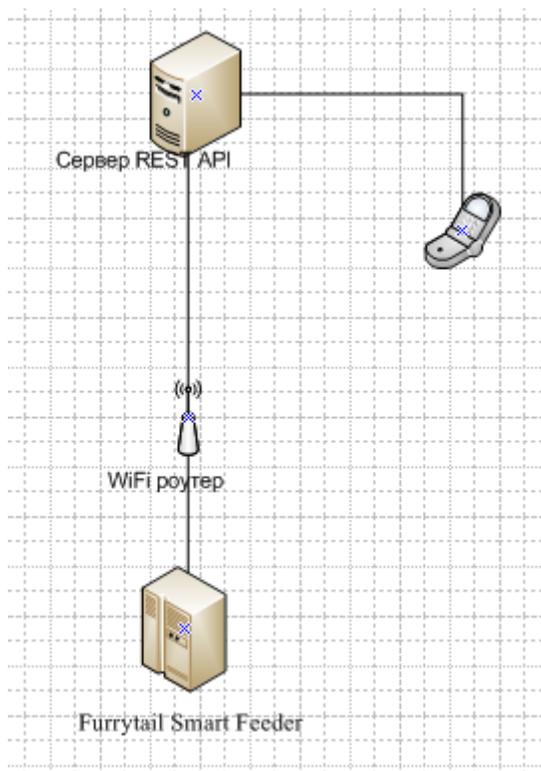


Рисунок 13 – структура кормушка на макро уровне.

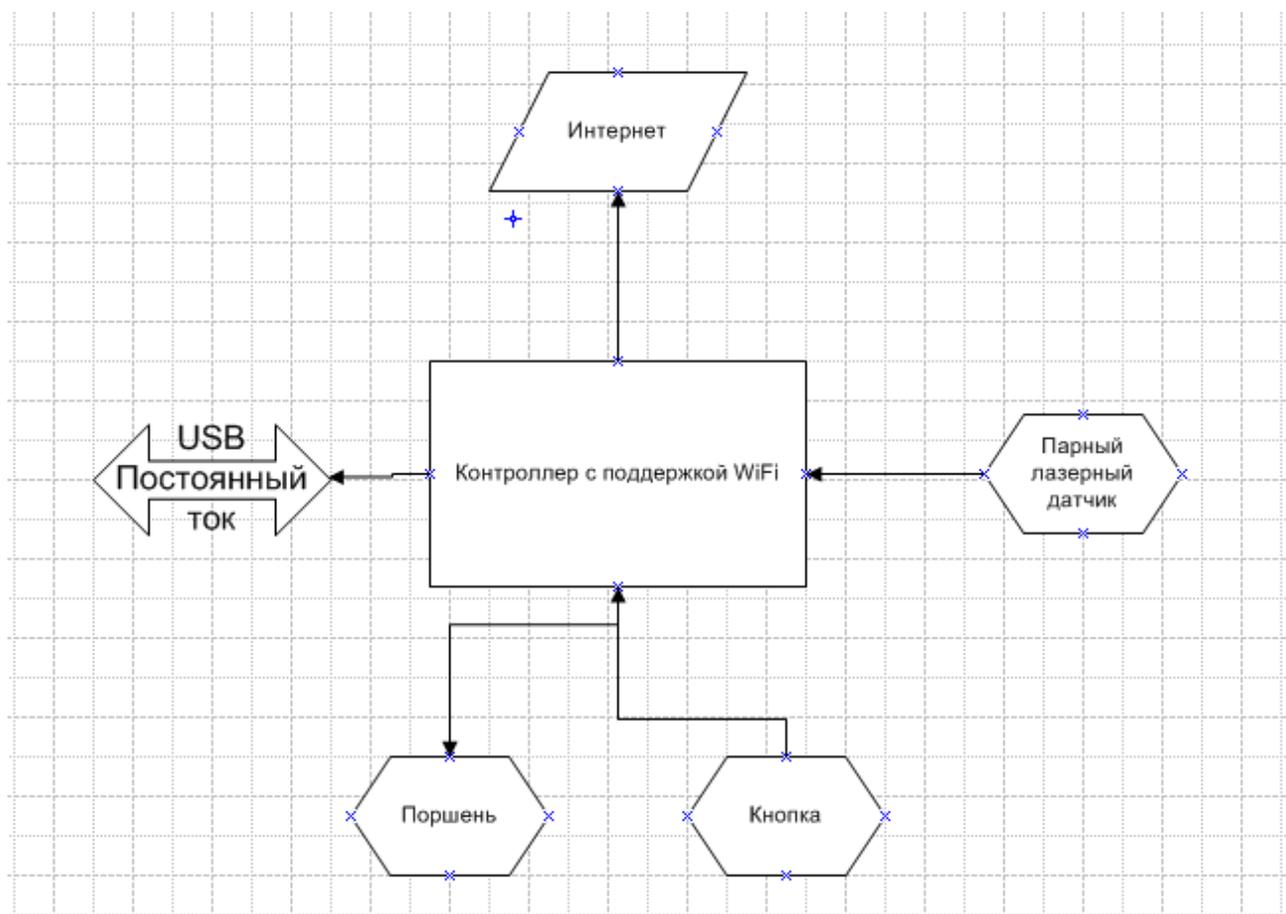


Рисунок 14 – структура кормушка на микро уровне.

Из представленных структур кормушки Furrytail Smart Feeder видно, что она работает от постоянного тока по кабелю USB, настройка может быть организована только через приложение на смартфоне, поддерживает WiFi, использует поршень для высыпания корма, реагирует на нажатую кнопку – открывая корм [8].

Умная кормушка **Petoneer Nutri Feeder** отличается от предыдущей только дизайном и тем, что приложение переведено на несколько языков. Но имеет еще

и отдельно поилку **Petoneer Fresco Ultra**, которая работает по принципу фонтана.



Рисунок 15 – Кормушка Petoneer Nutri Feeder.



Рисунок 16 – Поилка Petoneer Fresco Ultra.

Автоматическая кормушка для кота на Arduino сделана из консервных банок, Arduino UNO, кнопки и миски [9] энтузиастом. Работает по принципу – кот нажал кнопку – получил корм. Для использования корма нужно дрессировать кота, нет контроля за количеством корма.



Рисунок 17 – Кормушка для кота на Arduino.

1.3 АНАЛИЗ НЕДОСТАТКОВ АНАЛОГОВ

Любое устройство имеет недостатки, их необходимо учитывать перед созданием своего похожего устройства.

Кормушка Feed-Ex для сухого корма позиционирует себя как автоматическая кормушка, она не следует концепции «Интернет вещей», работает только от пальчиковых батареек, которые нельзя зарядить. Не смотря на то, что контейнер для корма довольно большой – нет контейнера и подачи воды. Интерфейс настройки очень сложен и не каждый быстро разберется, так как используется внутренний сервер. Запись голоса – входит в плюс, хоть и не обязательна.

Автоматическая кошачья кормушка для влажной еды – Feed and Go Automatic Feeder тоже не направлена на концепцию «Интернет вещей». Ее

контейнер имеет деление на ровные 6 разделов, больше или меньше сделать нет возможности, подачи воды так же нет. Камера и аудиозапись – это не необходимые устройства, но идут плюсом.

Кормушка SuperFeed CSF-3 - имеет маленький контейнер для корма, нет подачи воды, поддерживает только мелкий корм.

LUSMO Автоматический раздатчик еды для контроля веса – ограничен в размере корма, не имеет подачи воды.

Кормушка Furrytail Smart Feeder – не имеет подачи воды, язык только китайский, перевода нет, работает от постоянного тока по кабелю USB.

Кормушка Petoneer Nutri Feeder и поилка Petoneer Fresco Ultra имеют подачу воды, от предыдущей отличаются только внешним видом и интерфейсом мобильного приложения (и перевод есть).

Автоматическая кормушка для кота на Arduino – имеет маленький контейнер, нужна дрессировка кота, не связана с концепцией «Интернет вещей».

Из всех приведенных выше «умных кормушек» можно сделать вывод, что кормушка с кнопкой и ошейником не являются хорошей идеей, поскольку в таких случаях требует дополнительной дрессировки животных, а так же не получится кормление по времени и ни о какой статистике речи быть не может. Кормушка с кормом, но без воды не полноценна, поскольку домашние животные и пьют и едят, а подавать автоматически одно, но без другого – оставить животного с жаждой. Кормушки с поилкой отдельно – тоже плохая идея, поскольку на них нужны отдельные приложения и облачные вычисления.

Нужна кормушка, которая будет подавать и воду, и корм, по датчикам определять наполненность кормом и водой контейнеров, и мисок, синхронизироваться с сервером и мобильным приложением. Иметь разные настройки, отслеживать статистику, уведомлять о пустых контейнерах,

анализировать активность питомца, иметь разные режимы работы, включая статистические (самообучающаяся «умная» кормушка).

Стоит учитывать, что внутренний сервер с локальной сетью WiFi, как и отсутствие сети совсем – плохой вариант, так как концепция «Интернет вещей» предполагает выход в сеть, поэтому нужно разработать клиент-серверное приложение, где клиентом будут выступать сама кормушка и мобильное приложение.

Для внутренней сети есть способ вывода в интернет, но это требует знаний системного администратора как минимум. Не все готовы в этом разбираться, а так для тех, кто покинул дом на продолжительное время, важно, чтобы данные кормушки можно было отследить из любой точки мира.

Так же, «умная» кормушка должна работать как от электрической сети, так и от батареек (не пальчиковых, так как их надо менять), что гарантирует продолжительную работу кормушки даже с перебоями электричества, а при разряженной батарее и вовсе открывать выход корма на экстренные случаи.

Помимо вышесказанного – «умная» кормушка обязана выдавать советы по кормлению питомцев на основе данных о питомце.

2. РАЗРАБОТКА МОДЕЛИ, ПРОЕКТИРОВАНИЕ

Для реализации системы «умная» кормушка необходим следующий набор подсистем:

1. *HTTP-сервер Apache* с веб-интерфейсом в виде набора *PHP-скриптов*. С помощью веб-сайта обеспечивается общее управление системой, создание, конфигурация, управление и получение статистических результатов работы «умной» кормушки для домашних животных, а так же авторизации пользователей.
2. *MySQL* база данных, обеспечивающая хранение данных о пользователях, состояние кормушки, статистику работы.
3. *Клиентское приложение под мобильные платформы* – организация выполнения вычислений за счет ресурсов пользователя, получение аналитики от сервера и отправка команд на сервер.
4. *Графический интерфейс клиентского приложения* – настройка и контроль работы клиентского приложения, визуальное отображение авторизации в системе, получения задания и прогресса их выполнения;
5. *Веб-страницы* – удаленное управление или просмотр аналитики без использования мобильного устройства.
6. *Arduino Mega 2560* – главный контроллер «умной» кормушки
7. *Node MCU Amica на ESP8265* – WiFi модуль для передачи данных от Arduino к веб-серверу по REST API.
8. *Датчики, дисплей, кнопки, защита батареи, батарея и серво приводы* – отвечающие за реакцию на состояние контейнеров, мисок и питомца.

2.1. ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

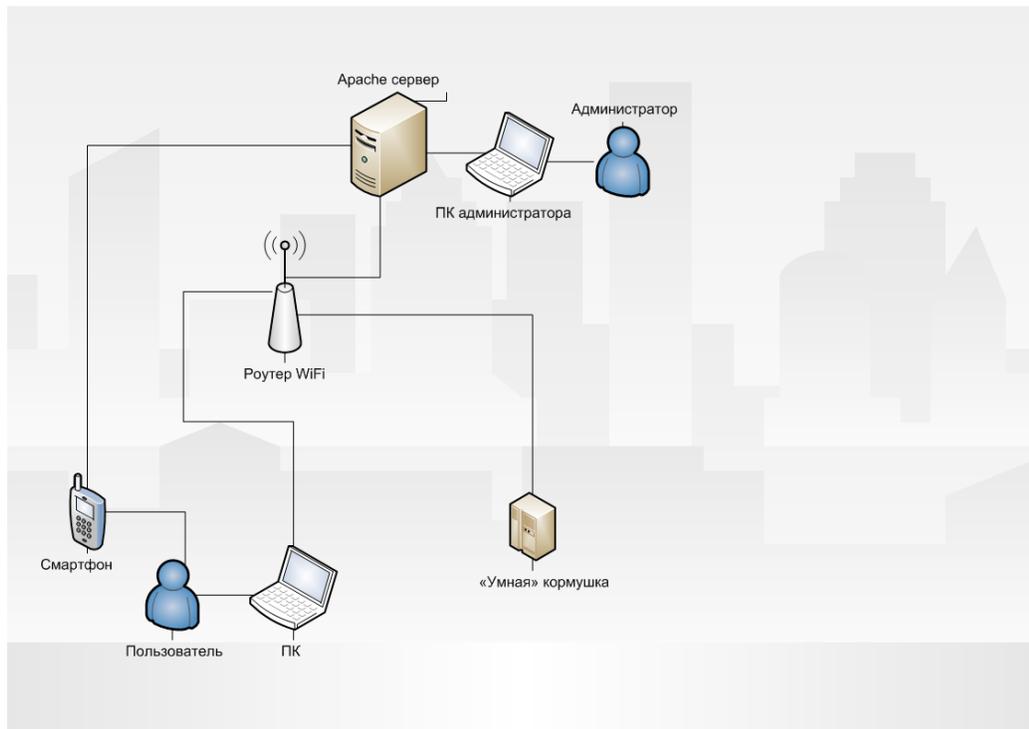


Рисунок 18 – Схема вариантов использования системы «умная» кормушка для домашних животных

На рисунке 18 изображена диаграмма вариантов использования системы «умная» кормушка для домашних животных. Можно выделить следующих ключевых актеров, взаимодействующих с системой:

- *Администратор сервера*: как правило, прикладной программист, работающий над сервером отображения интерфейса управления и статистики «умной» кормушки, а так же следящий за его стабильностью.
- *Apache сервер*: основной сервер управления «умной» кормушкой и отображающий информацию, поступающую от нее.

- «Умная» кормушка: устройство, которое выполняет команды сервера, анализирует датчики и отправляет состояние и статистику на сервер Apache.
- *Пользователь*: человек, который может с телефона или с компьютера (из своей подсети WiFi или нет) контролировать свою «умную» кормушку и получать данные от нее.

Рассмотрим прецеденты использования разрабатываемой системы.

Пользователю необходимо *авторизоваться*, чтобы иметь возможность настраивать «умную» кормушку и получать от нее данные: для этого он вводит в соответствующем интерфейсе системе свои имя учетной записи и пароль, которые предоставляются вместе с умной кормушкой.

После процедуры авторизации пользователь системы может перенастроить кормушку под себя (или своего питомца), получать уведомления об опустошении контейнеров или просто отслеживать статистические данные о ней и питомце.

Все данные пользователя сохраняются на сервере в MySQL.

2.2. НЕФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

В системе должна быть предусмотрена защита от следующих типов угроз безопасности:

- Умышленное переполнение базы данных сервера.
- Кража данных учетной записи с сервера.
- Перехвата передаваемых данных учетной записи.
- Кража файлов участников сети.
- Злоупотребление ресурсами участников сети.

Проблемы в пунктах 2,3,4 решаются путем создания шифрования данных, а так же шифрования SSL.

Безопасность данных в открытых информационных сетях, таких как Интернет, всегда будет источником серьезного беспокойства для разработчиков и клиентов. Поэтому для любого используемого продукта крайне важно создать безопасную среду исполнения.

Протокол SSL (Secure Socket Layers – протокол защищенных сокетов), совместно разработанный Netscape Communications и RSA Data Security, позволяет эффективно обеспечить такую безопасность. Протокол SSL обеспечивает безопасность, аутентификацию на базе сертификатов и согласование безопасности по установленному сетевому соединению, поэтому множество компаний и продуктов приняли SSL в качестве коммуникационного протокола[10].

Проблемы в пунктах 1,5 решаются с помощью анализа журналов системы контроля версий, поиска уязвимостей и установки stable релизов ПО.

2.3. ВНЕШНИЙ ВИД И СТРУКТУРА «УМНОЙ» КОРМУШКИ

Внешний вид и структура «умной» кормушки представлена на рисунке 19.

Под цифрой 1 показан принцип работы контейнера воды, который состоит из 5 литровой бутылки, вставленной в специальную воронку. В верхней части воронки установлен датчик расстояния (зеленый), который определяет наполненность контейнера воды. Бутылка съемная, после вложения ее в воронку – полость воронки становится герметичной, что не позволяет выходить воде за уровень высоты миски.

Под цифрой 2 показан контейнер с кормом, он так же имеет датчик расстояния (зеленый) для определения наполненности, но в конце воронки

располагается еще сервопривод с диском, при вращении которого открывается или закрывается высыпание корма.

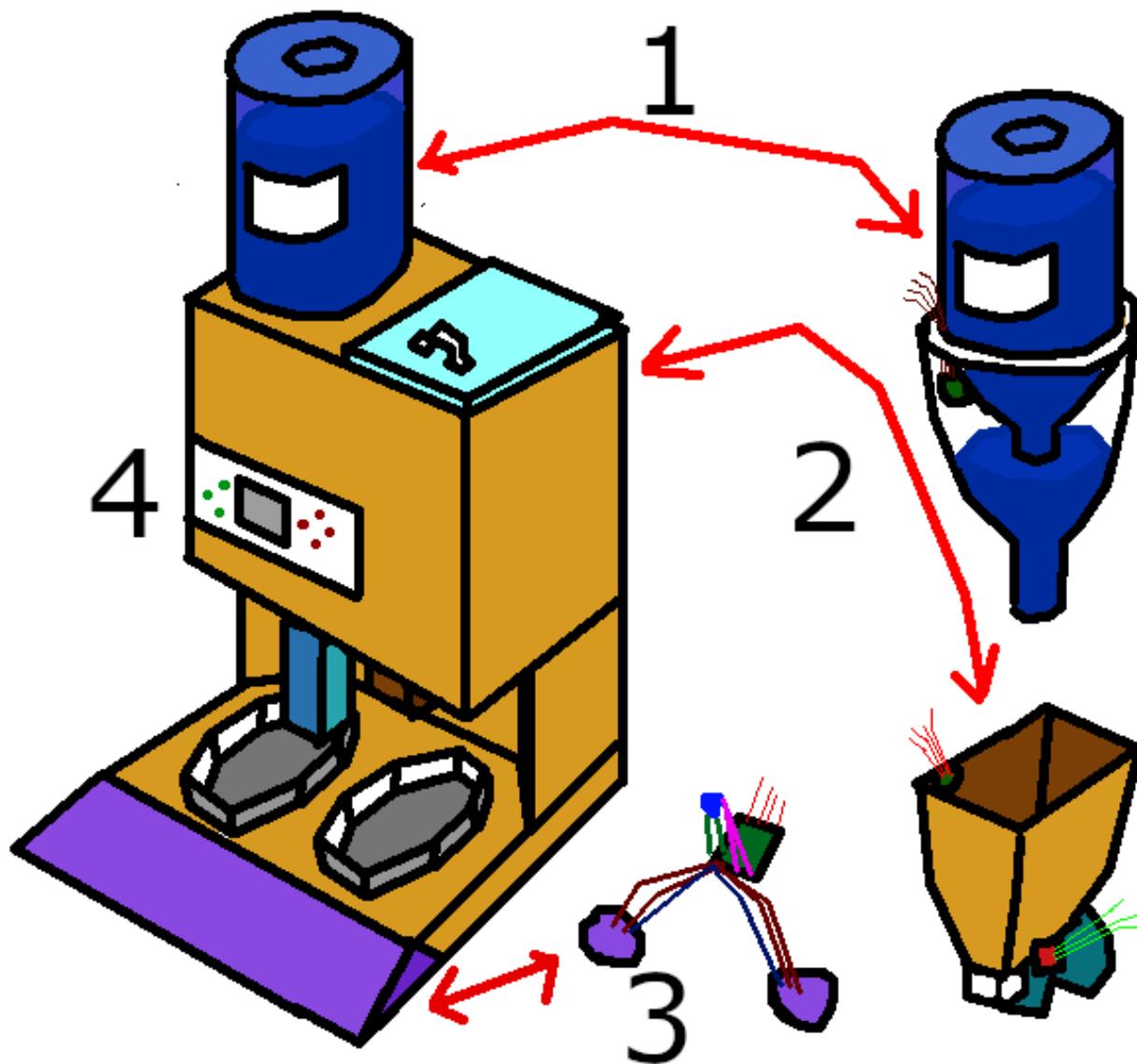


Рисунок 19 – Внешний вид и структура «умной» кормушки

Под номером 3 показаны 2 тензодатчика (датчика весов, фиолетовые), которые соединены с АЦП (аналого-цифровым преобразователем, зеленым) проводами (красными, синими и розовыми) и резисторами (зелеными в виде проводов и синим).

Контейнер и с водой и с кормом выходят в миски, на которые направлены датчики движения и 2 датчика расстояния. Данные датчики определяют активность питомца.

Под номером 4 располагается Nokia LCD дисплей и 7 кнопок управления кормушкой offline (без сети). Данная панель позволяет настроить данные авторизации на сайте управления, имя и пароль роутера WiFi, а так же небольшие локальные параметры работы.

Красные кнопки отвечают за ввод текста, а именно: левая – стирает текст, правая ставить пробел, верхняя выбирает символ на 1 выше по ASCII таблице, нижняя выбирает на 1 символ ниже.

Зеленые кнопки отвечают за подтверждение (или ввод символа, нижняя), отказ (средняя) и меню (верхняя).

Кормушка имеет 3 режима управления:

- 1 Анализа и реакций – кормушка анализирует датчики, отправляет данные в сеть, при необходимости подает корм в миску.
- 2 Меню – в этот момент не отправляются аналитические данные на сервер, но кормушка может подавать корм при необходимости. Данное состояние вызывается зажатием кнопки «меню» и позволяет настроить кормушку, ввести логин и пароль, а так же SSID и пароль от WiFi сети. Если нажать «меню» еще раз – то кормушка вернется к 1 режиму.
- 3 Ввод логина и пароля или SSID и пароля сети. После ввода данным – нужно нажать кнопку «меню» для подтверждения или кнопку «отказ» для отмены ввода.

3. РЕАЛИЗАЦИЯ ПРОТОТИПА «УМНОЙ» КОРМУШКИ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ

3.1. СТРУКТУРА И ПРИНЦИП РАБОТЫ УСТРОЙСТВА «УМНОЙ» КОРМУШКИ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ

Структура «умной» кормушки для домашних животных состоит из нескольких отдельных макромодулей и внутренних микромодулей.

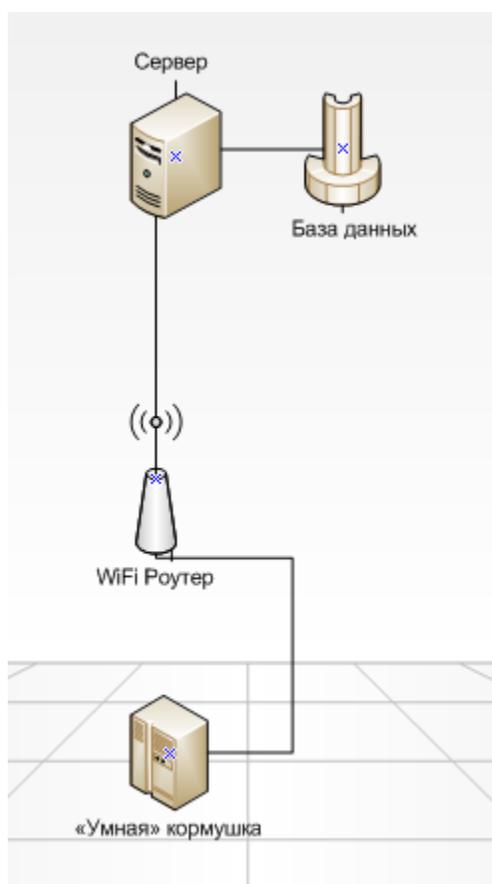


Рисунок 20 – Макромодульная часть структуры «умной» кормушки

Макромодули представляют собой целостные устройства, взаимодействующие между собой образуя систему (концепция «Интернет вещей»). На рисунке 20 представлены следующие макро модули:

– «Умная» кормушка – само устройство для подачи корма и воды животным, которое обращается к серверу для предоставления статистических данных и получения информации о режиме работы.

– WiFi роутер – беспроводной способ соединиться «умной» кормушке и серверу и начать обмен данными. Как известно, одним из столпов, на которых зиждется успех IoT, является потребность подключить множество IoT-устройств к остальному миру через Интернет. Можно с уверенностью предположить, что основой для этого станет именно беспроводная связь. Вопрос остается в том, какая из технологий наиболее подходящая? Сейчас существует много вариантов, каждый из которых предлагает различные способы установления соединения для приложений IoT. Наиболее востребованными являются Wi-Fi, Bluetooth и их разновидности, а также LTE на базе сотовой связи [9].

– Сервер – некий компьютер или облако, находящееся далеко от «умной» кормушки и соединенная с другими сетями и роутерами. Принимает статистику и отправляет параметры, так же предоставляет возможность пользователям отслеживать статистику и менять настройки «умной» кормушки.

– База данных – некое отдельное от сервера или находящееся на нем место для хранения полученных и измененных данных.

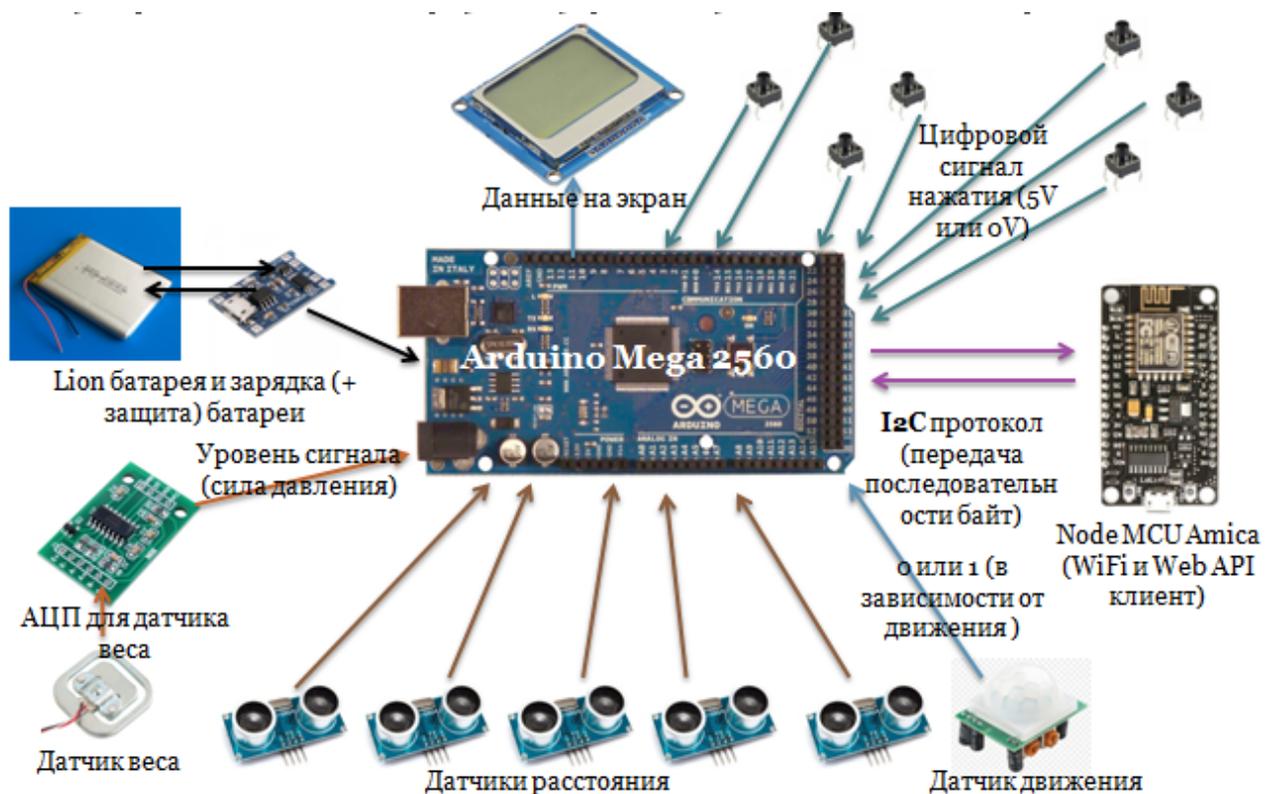


Рисунок 21 – Микромодульная часть структуры самого устройства «умной» кормушки

На рисунке 21 изображена микромодульная структура самого устройства умной кормушки. В нее входят:

- Arduino Mega 2560 – основной микроконтроллер и анализатор датчиков, который обменивается данными с Node MCU Amica по I2C протоколу для передачи данных по WiFi или получения ответа от сервера.

- Node MCU Amica – WiFi модуль, задача которого – соединиться с сервером, передавая данные от Arduino Mega или получения данных, которые отправятся на Arduino Mega по протоколу I2C.

- 5 датчиков HC-SR04 – датчики расстояния, определяющие, сколько корма в контейнерах, полны ли миски, ест ли питомец в данный момент. Затем данные отправляются на Arduino Mega для анализа и передачи или не передачи данных на Node MCU Amica. Данный набор датчиков позволяет отследить как

можно больше статистики, а не только время кормежек и наполненности контейнера с кормом.

- Датчик движения – позволяет определить, что питомец активен.

- Серво привод, отвечающий за открытие и закрытие выводов корма в миску. При открытии – корм засыпается в миску из контейнеров. Помимо корма идет еще подача воды, но она не управляется. Вода доступна всегда. В рассмотренных ранее кормушках была подача только корма.

- Зарядка и Lion батарея – источники питания для обеих плат, которые действуют по принципу – если нет электричества дома, то в батарее еще на какое-то время хватит. В рассмотренных кормушка в главе 2 кормушка работала либо только от сети, либо только от пальчиковых батареек, что могло привести к отказу кормушки и голодовке питомца.

- Защита батареи HW-168 – позволяет заряжать устройство от usb и защищать от перезаряда.

- Тензодатчик – измеряет вес питомца. Предел датчика до 50 кг.

- АЦП тензодатчика – микросхема, отвечающая за определение веса и передачу этих данных главному микроконтроллеру.

Целесообразность микромодульной структуры самого устройства описана дальше.

Интерфейс I2C (или по другому IIC) — это достаточно широко распространённый сетевой последовательный интерфейс, придуманный фирмой Philips и завоевавший популярность относительно высокой скоростью передачи данных (обычно до 100 кбит/с, в современных микросхемах до 400 кбит/с), дешевизной и простотой реализации [11].

Arduino MEGA 2560 имеет цифровых 53 вывода, что позволяет подключить большое количество датчиков.

Использование Arduino MEGA 2560 дало возможность сделать по-настоящему большую и сложную микроконтроллерную систему. Например, есть проект, который получил поддержку в РФ и активно развивается – это Arduino Mega Server. Микроконтроллер настолько мощный, что может стать целым сервером для интернета сайтов или облака.

Единственное ограничение на таком сервере – это объём памяти, ведь в качестве накопителя можно использовать micro SD-карты памяти, а Ethernet поддерживает максимальный объём памяти 32 гб [12].

Сердцем платы NodeMcu V2 является недорогой чип ESP-12E, в котором встроен микроконтроллер ESP8266 с тактовой частотой 80 МГц (можно разогнать до 160 МГц) и флеш память на 4 Мб. ESP8266 включает в себя приемопередатчик Wi-Fi b / g / n, поэтому можно не только подключаться к сети Wi-Fi но и создавать собственную сеть [13].

Данные между Arduino Mega и Node MCU по I2C, а так же между Node MCU и сервером передаются посредством последовательности символов l,r,x,w,~,1,«.» . Тильда – означает отсутствие события, 1 – наличие, больше 0.00 – вес, l – логин, r – пароль, w – SSID имя WiFi сети, x – SSID пароль .

Тильда используется исключительно потому, что I2C протокол в Arduino передает символ под кодом 0x00 как «ничего» и потому последовательность символов сдвигается в левый край.

Например:

d~~~~~0.00~~~~ – ничего не происходит, контейнеры с кормом и водой полные, миска полная, питомец не ест и не пьет, не активен, вес не измеряется из-за отсутствия веса.

d1~~~~~0.00~~~~ – корм в контейнере закончился.

d~1~~~~~0.00~~~~ – закончилась вода в контейнере.

d11~~~0.00~~~ – корм и вода в контейнерах закончились
соответственно.

d~1~~~0.00~~~ – миска с кормом пуста.

d~~~~10.00~~~ – питомец активен.

d~~~~3.77~~~ – питомец весит 3.77 Кг.

d~~~~0.001~~~ – питомец ест.

d~~~~0.00~1~~ – питомец пьет

ltest~~~~~ - логин test.

pctest~~~~~ - пароль test.

wtest~~~~~ - SSID имя test.

xtest~~~~~ - SSID пароль test.

Все эти события могут быть присланы все одновременно. Строка может
иметь такой формат d 1111~~0.99111~~ или d1~~1~~7.511~1~~ и т.д.

Помимо передачи данных из Arduino Mega 2560 в Node MCU Amica,
передача ведется в обратном направлении.

В обратном направлении данные передаются по одному символу,
например: d – запрос данных; w – запрос SSID имени; x – запрос пароля SSID; p
– запрос пароля для сервера; l – запрос логина для сервера; 0 – включить режим,
при котором корм засыпается по мере опустошения миски; 1 – включить режим
блокировки (меняется на 0 в зависимости от времени); 2 – включить режим, при
котором корм подается только когда питомец находится у миски и миска
пустая.

Сервер возвращает 2 символа m и число от 0 до 2, в зависимости от того,
какой режим был выбран в приложении Android. Например: m0 – включить в
кормушке обычный режим; m1 – не подавать корм, пока не вернется m0 или m2
– подавать корм по приходу питомца, но если миска пустая.

3.2. СТРУКТУРА И ПРИНЦИП РАБОТЫ СЕРВЕРНОЙ ЧАСТИ

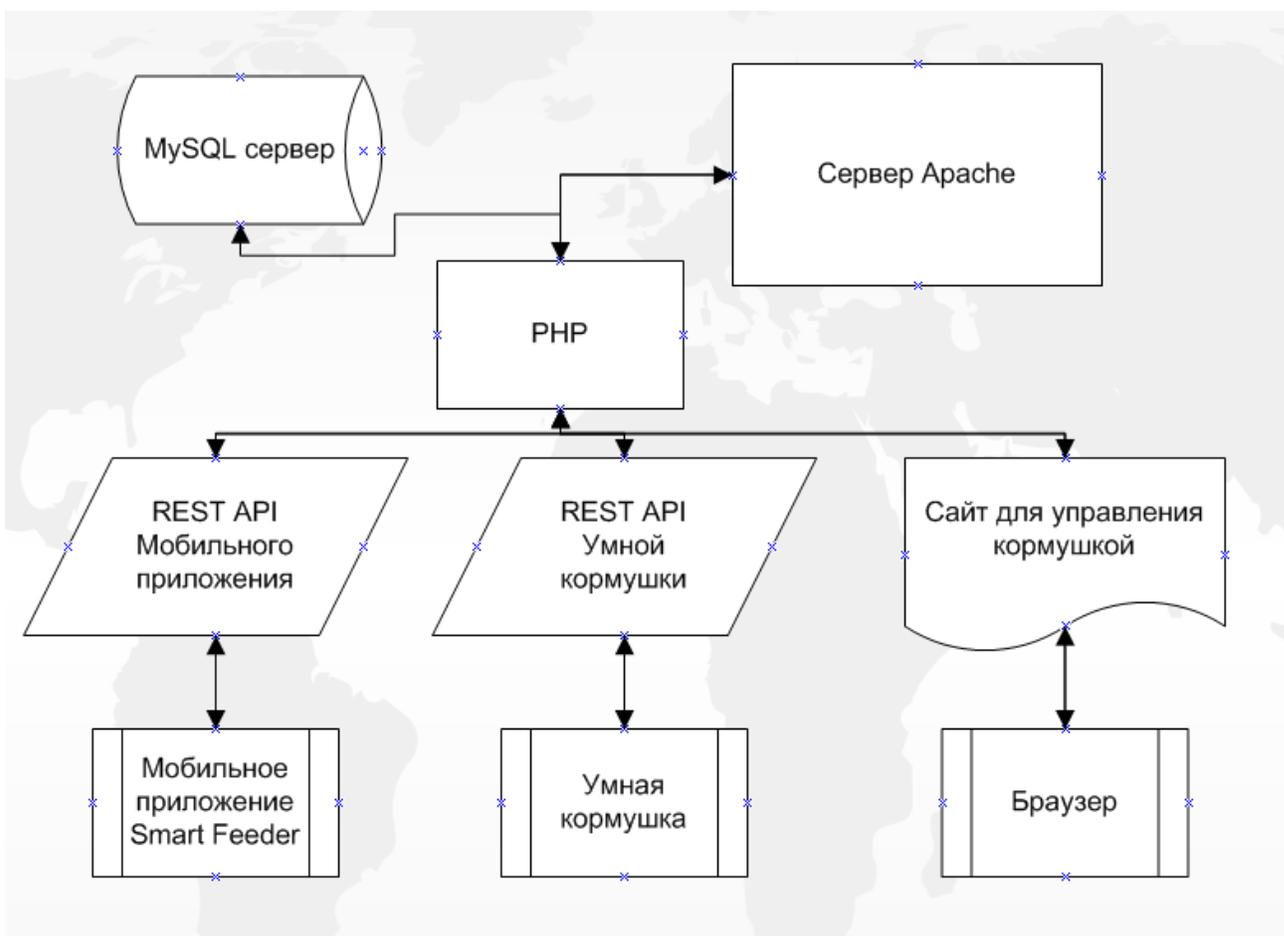


Рисунок 22 – Микромодульная часть структуры серверной части

На рисунке 22 представлена микромодульная структура серверной части. В ее состав входят:

– Сервер Apache – веб-сервер, перенаправляющий любые запросы к PHP. Его преимущество перед IIS очевидно, а именно – быстрая реакция на запросы к серверу.

– PHP – сервис для генерации веб-страниц и ответов REST API, обращающийся к MySQL базе данных. В отличие от ASP.NET MVC не требует перекомпиляции и не тянет за собой гору кода, который не понадобится в проекте

– MySQL – база данных, хранящая все записанные или измененные в ней данные.

– Сайт (авторизация, управление умной кормушкой и отображение данных). После авторизации открывается отображение данных с возможностью изменения настроек «умной» кормушки.

– Браузер – программа, отображающая PHP страницы (html). Не все указанные ранее аналоги поддерживали веб-сервер для отображения данных.

– Мобильное приложение – приложение, работающее с REST API и имеющая функционал как сайт, но удобнее.

Сочетание мобильного приложения и веб-сервера делает данные о кормушке легко доступными для разных типов пользователей.

– REST API – протокол GET и POST запросов к серверу с целью получить данные или внести изменения. REST API устройства и мобильного приложения располагаются в одном файле `get_response.php`. В приложении А есть ссылка на все исходные коды «умной» кормушки, включая REST API.

Причины данного состава микромодульной структуры серверной части описана ниже.

Веб-сервер Apache – HTTP сервер с открытыми исходными текстами для современных операционных систем, включая UNIX и Windows. Apache обладает обширными возможностями конфигурации, является очень производительным, поддерживает все известные протоколы и используется для организации большинства веб-серверов в мире. Существуют локализации сервера для различных языков, в том числе и для русского.

Согласно данным исследовательской компании Netcraft, ведущей мониторинг глобальной сети на протяжении последних 15 лет, по данным на 1 декабря 2008 г. Глобальная сеть насчитывала 186 727 854 сайта. Лидером рынка серверов является открытый сервер Apache, на долю которого приходится более

половины всех работающих сайтов - 51,24% [15]. Выбор веб-сервера Apache очевиден.

PHP: Hypertext Preprocessor (изначально Personal Home Page Tools, «Инструменты для создания персональных веб-страниц») — один из лидирующих языков современной веб-разработки. Его отцом считается датский программист Расмус Лердорф, который в 1994 году создал набор скриптов на Perl — ту самую «персональную домашнюю страницу», которая легла в основу PHP.

Со временем к работе над ним подключились разработчики со всего мира. PHP — один из старейших языков в рамках open source проекта. Сейчас его поддерживает и разрабатывает группа энтузиастов во главе с компанией Zend Technologies. Ей руководят Зеев Сураски и Энди Гутманс: в 1997 году они создали третью версию PHP и активно развивают язык по сегодняшний день [16].

Язык PHP интерпретируемый, гибкий, активно поддерживаемый, совместимый с Apache и один из самых популярных языков автоматизации веб-страниц.

База данных MySQL — это самая популярная в мире база данных с открытым кодом. Благодаря своей проверенной производительности, надежности и простоте использования база данных MySQL наиболее часто используется для веб-приложений на таких ресурсах, как Facebook, Twitter, YouTube и все пять из пяти лучших веб-сайтов*. Кроме того, ее особенно часто выбирают в качестве встроенной базы данных, распространяемой тысячами поставщиков программного обеспечения и производителей оборудования [14].

Интернет Вещей — одна из тенденций в разработке мобильных приложений.

Различные устройства связываются между собой, образуя сети, которые объединяются друг с другом и с Интернетом. IoT проник практически во все сферы жизни, и пока наибольшую актуальность он обрел в домашнем быту, медицине и промышленности, но область его влияния стремительно растет. Многие компании интересуются этим направлением и уже серьезно занимаются его разработкой, и к ним относятся такие гиганты, как Google, Samsung, Apple, LG, Intel, Qualcomm и многие другие [17].

3.3. МОБИЛЬНОЕ ПРИЛОЖЕНИЕ

Для полноценного взаимодействия с «умной» кормушкой необходимо ПО, которое уведомляет об опустошении контейнеров, разрядке аккумулятора, отображает статистику, советы, состояние и позволяет изменить режим работы «умной» кормушки.

Для достижения поставленной задачи разработано мобильное приложение «Nekitgam SmartFeeder», которое удовлетворяет данным запросам.

Ниже представлен интерфейс мобильного приложения на рисунках 23, 24 и 25.

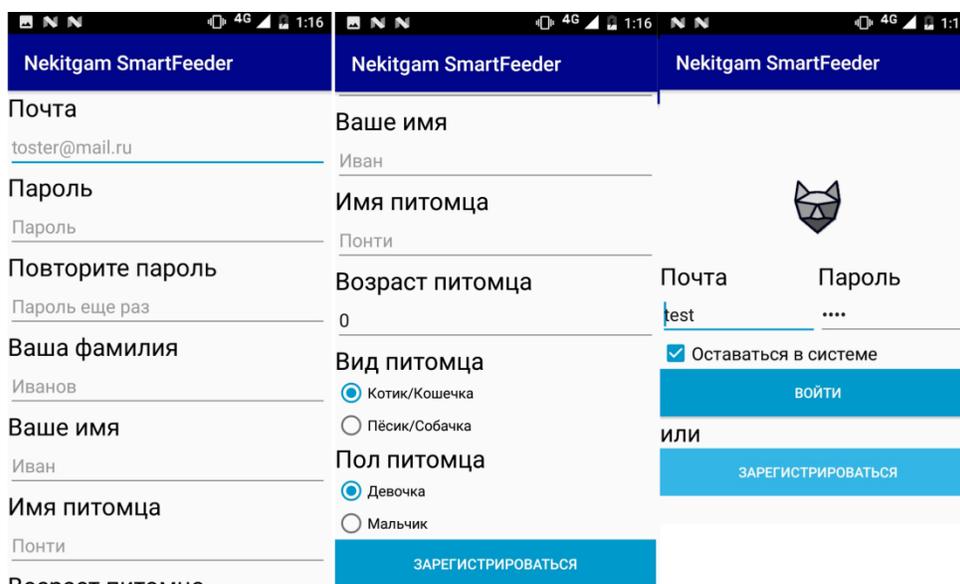


Рисунок 23 – Активити входа и регистрации.

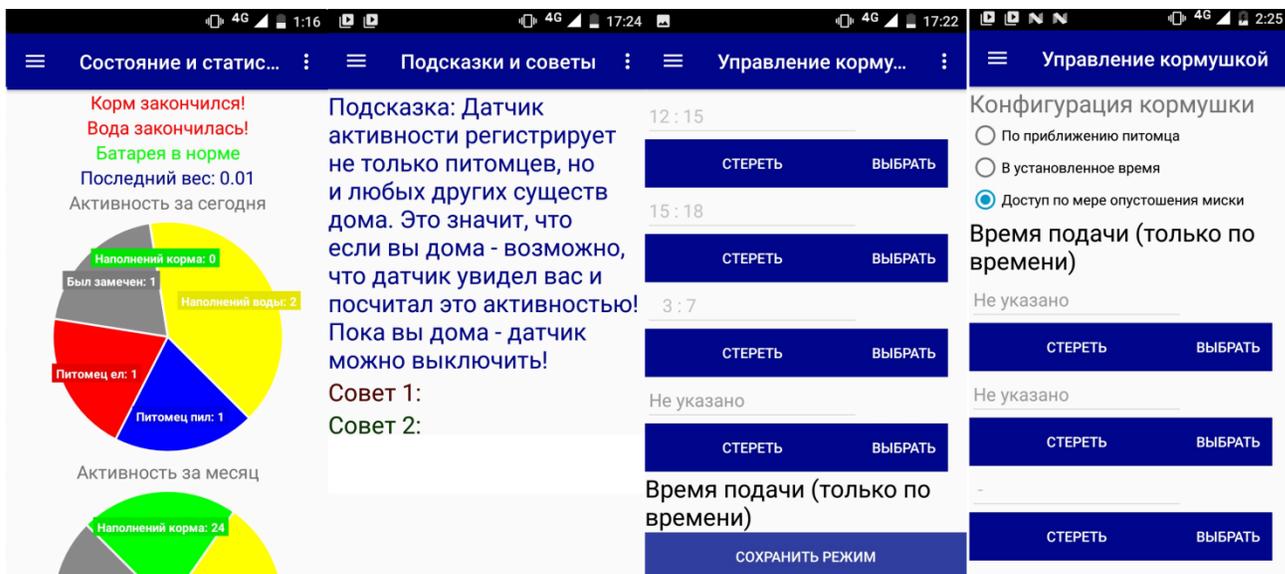


Рисунок 24 – Внешний вид главных активити приложения.

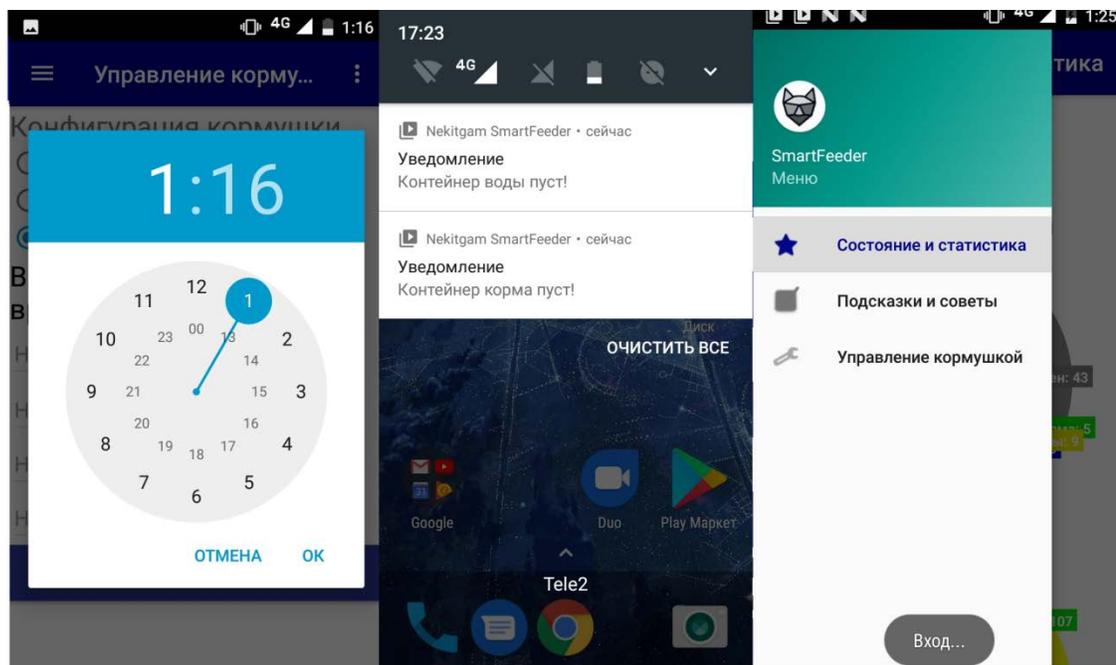


Рисунок 25 – Уведомления. Диалог выбора времени работы кормушки и меню.

Приложение «Nekitgam Smart Feeder» работает в двух потоках одновременно.

Первый поток является основным для приложения, в котором происходит авторизация, регистрация, отображение и ввод данных. Он закрывается как только происходит выход из приложения.

Второй поток – это сервис Android, который работает как во время работы приложения, так и фоном вне приложения. Его задача – запрос каждые 5 минут данных с сервера и уведомление пользователей о проблемах (если в кормушке пустые контейнеры или разряженный аккумулятор). Уведомления представлены на рисунке 25.

Для отображения статистики используются круговые диаграммы (в Android называемые Toast, рис. 24), которые делятся на треугольники с названием поля и количеством данных записей в БД (в Android называются Slices)

3.4. ПРОТОТИП «УМНОЙ» КОРМУШКИ

В результате сборки внешний вид прототипа «умной» кормушки для домашних животных приобрел следующий вид (рис 26):



Рисунок 26 – Прототип «умной» кормушки.

Прототип «умной» кормушки имеет отличие от макета наличием выпуклой части с фронтальной стороны, внутри которой и находятся все микроконтроллеры и модули.

Корм насыпается за счет быстрого движения сервопривода на 15 градусов в одну и обратную сторону, что позволяет насыпать корм медленно, и не рассыпая вокруг. Корм из контейнера напрямую давит на движущийся элемент сервопривода малым весом (за счет сужения радиуса контейнера к концу).

Вода не наливается, а держится простым физическим явлением: полная 5-ти литровая давит на проем. Проем под весом сжимается со всех сторон. Тем самым образуется герметичное пространство, не позволяющее проникнуть воздуху в полость контейнера никаким образом, кроме как из миски, которая находится выше, чем горло от выхода контейнера, что заставляет контейнер тянуть воду обратно и тем самым не выливать за края миски.

Таким образом – вода не выливается за миску без какого-либо перекрытия.

Зарядка «умной кормушки» производится через micro usb выход, что позволяет использовать любой шнур от телефона Android. При этом – умная кормушка должна быть всегда на зарядке. Аккумулятор встроен только для экстренных случаев, защищен от перезарядки и способен работать сутки без подзарядки.

3.5. ОПИСАНИЕ ДАННЫХ

Для хранения данных используется сервер базы данных MySQL, и в нем создана база данных «smart_feeder», столбцы которой представлены на рисунке 27.

Имена строк, их типы и значения:

Id (целочисленный) – ключевой идентификатор записи.

FullName (строковой) – полное имя владельца «умной» кормушки.

Email (строковой) – почта.

EncryptedCode (строковой) – зашифрованный код пользователя, по которому и происходит авторизация.

None_Food и None_Water (логический) – становятся True если контейнеры с кормом или водой пусты.

Mode (целочисленный) – режим работы «умной» кормушки.

User_Id (целочисленный) – ссылка на Id пользователя.

Date (время и дата) – дата события, записанного в статистику.

Full_Food и Full_Water (логический) – контейнеры были заполнены после опустошения.

Pet_Water и Pet_Eat (логический) – событие, когда питомец ест или пьет.

Clear_Food(логический) – событие, когда в миске закончился корм.

Weight (с плавающей запятой) – вес питомца.

Weigth_max (с плавающей запятой) –максимальный вес.

Weight_min (с плавающей запятой) – минимальный вес.

Hint (многострочный) – совет для питомца.

Pet_Type (логический) – вид: кошка или собака.

Pet_Sex (логический) – пол питомца.

Pet_Age (логический) – возраст питомца.

PetActive (логический) – питомец двигался возле кормушки и кормушка это зарегистрировала.

Low_Battary (логический) – становится true, когда батарея кормушки садится.

Time,Time1,Time2,Time3 (строковой) – поля, необходимые для работы режима «умной» кормушки «по времени».

Таблица соответствует правилам целесообразного проектирования таблиц, представленных в литературном источнике «Проектирование таблиц для базы данных»:

- Всегда должен быть уникальный идентификатор.
- Используется связь «один ко многим» для таблиц «statistic» и «users», при которой в «statistic» помещаются множество записей с указанием Id пользователя, которому событие принадлежит.
- Все данные разделены на отдельные столбцы.
- Некоторые данные имеют запрет на значение NULL.
- Имена таблиц отражают их назначение.
- Таблицы не имеют сильно сложную структуру, что ускорит загрузку и нахождение данных в БД [18].

Таблица Users – отвечает лишь за авторизацию и состояние кормушки.

Таблица Statistics – отвечает только за статистику событий и его время происхождения.

Таблица Hints - нужна для интерфейса пользователя и отвечает за советы пользователю умной кормушки в зависимости от веса, возраста, пола и вида питомца. Если минимальный и максимальный вес равны 0, тогда данное поле считается подсказкой, а если хотя бы один не равен нулю – считается советом.

На рисунке 27 данные, которые необходимы таблице «hints» указаны серой линией условно, поскольку аргументы для выборки в запросе к данной таблице приходят с мобильного приложения.

База данных не перегружена кучей таблиц, трех более чем достаточно.

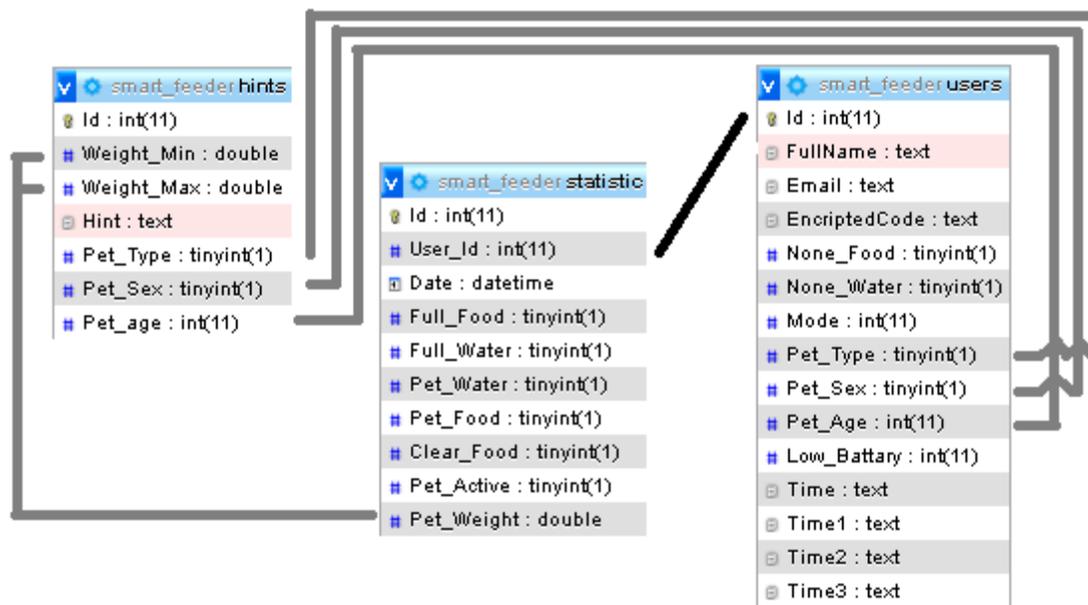


Рисунок 27 – Схема разработанной базы данных авторизации и статистического анализа

4. ТЕСТИРОВАНИЕ, ОТЛАДКА, ЭКСПЕРИМЕНТЫ

4.1. ТЕСТИРОВАНИЕ ДАТЧИКОВ И МОДУЛЕЙ

Для предоставления пользователю точной информации – в первую очередь нужно откалибровать датчики.

В состав «умной» кормушки входят такие датчики, как: тензодатчик на 50кг. (весовой), расстояния (анализаторы состояния кормушки) и датчик движения (выявляющий активность питомца).

При тестировании тензодатчика, поначалу были обнаружены проблемы, которые впоследствии были устранены. Датчик не стабильно измерял вес и через время переставал работать.

Тензодатчик подключался при помощи аналого-цифрового преобразователя HW-29, двух резисторов на 100Ком и подстроечного резистора на 500Ком и при минимальной влаге (наличия клея) выдавал помехи и через время переставал передавать данные. Решение данной проблемы лежит в просушивании датчика и изделия, к которому он крепится.

Во время тестирования датчиков расстояния были обнаружены помехи замера воды и корма в контейнерах (причиной тому было стремящееся углубление контейнера в середину), но проблему удалось решить при помощи анализа со стороны сервера. Сервер на PHP коде замерял данные в 6 этапов и если данные расходились в 50% (приходило 3 положительных и 3 отрицательных значения), то данные учитывались как «пустой контейнер», если было иначе – как полная, поскольку корм или вода перекрывают нижнюю часть контейнера, которая и стремится в центр. С остальными датчиками расстояния проблем не случалось.

Датчик движения временами не передавал данные на сервер (данные на сервер отправляются раз в секунду, а состояние датчиков за это время могло

меняться, и датчик движения при обнаружении движения – обнуляет состояние). Данная проблема решилась увеличением времени паузы после обнаружения до 500 миллисекунд.

4.2. ТЕСТИРОВАНИЕ ПЕРЕДАЧИ ДАННЫХ МЕЖДУ МИКРОКОНТРОЛЛЕРАМИ

Как упоминалось в главе 3.1 – данные между платами передаются как последовательность символов `d~~~~~0.00~~~~`, где самый первый символ – это тип передаваемых данных. Так как данные об имени WiFi сети, пароле, логине и пароле сервера вводятся в Arduino Mega 2560 и сохраняются в его же еергом память, то при включении кормушки WiFi модулю приходится запрашивать эти данные. И обнаружилось, что если этих данных в еергом памяти Arduino не было – кормушка не реагировала на кнопки и не позволяла эти данные вписать.

Данная проблема решилась увеличением интервала запроса этих данных до 1 секунды.

Второй проблемой было появление в конце передаваемой строки не известных знаков (от 127-255 по таблице ASCII), и из-за этого временами данные на сервер не передавались.

Решилась проблема заменой типа данных String на `char*`.

4.3. ТЕСТИРОВАНИЕ АВАРИЙНЫХ СИТУАЦИЙ

В «умной кормушке» есть протокол действий на случай отключения интернета, сети и разрядки батареи. При заряде батареи меньше чем на 10% - кормушка должна открывать корм (с целью выживания питомца, если хозяев дома нет продолжительное время) и передать уведомление хозяевам на мобильный телефон и на сайт. При отключении интернета в умной кормушке

существуют режим аварийной работы, так что режим работы кормушки не нарушится.

Данные ситуации тест прошли. Кормушка при 5% заряда батареи открывала корм, а отсутствие сети никак не влияло на работу кормушки, кроме как передачу статистических данных. Так же, аккумулятор работал сутки без зарядки, что отслеживалось по базе данных (рис. 28).

Id	User_Id	Date	Full_Food	Full_Water	Pet_Water	Pet_Food	Clear_Food	Pet_Active	Pet_Weight
776	1	2020-05-26 05:11:21	0	0	0	0	0	0	3.54
777	1	2020-05-26 05:12:01	0	0	0	0	0	0	3.61
778	1	2020-05-26 05:13:02	0	0	0	0	0	0	3.63
779	1	2020-05-26 05:13:23	0	0	0	0	0	0	3.59
780	1	2020-05-26 05:14:03	0	0	0	0	0	0	3.6
781	1	2020-05-26 05:14:23	0	0	0	0	0	0	0
782	1	2020-05-26 05:14:43	0	0	0	0	0	0	3.69
783	1	2020-05-26 05:15:04	0	0	0	0	0	0	0
784	1	2020-05-26 05:15:44	0	0	0	0	0	0	3.73
785	1	2020-05-26 05:16:05	0	0	0	0	0	0	3.58
786	1	2020-05-26 05:17:05	0	0	0	0	0	0	3.66
787	1	2020-05-26 05:18:26	0	0	0	0	0	0	3.7
788	1	2020-05-26 05:19:06	0	0	0	0	0	0	3.7
789	1	2020-05-26 05:19:27	0	0	0	0	0	0	0
790	1	2020-05-26 05:20:07	0	0	0	0	0	0	3.61
791	1	2020-05-26 05:20:27	0	0	0	0	0	0	0
792	1	2020-05-26 05:21:08	0	0	0	0	0	0	3.59
793	1	2020-05-26 05:24:10	0	0	0	0	0	0	3.73

Рисунок 28 – Записи с «умной» кормушки ночью в базу данных, без непосредственного контроля.

4.4. ТЕСТИРОВАНИЕ ПЕРЕДАЧИ ДАННЫХ НА СВЕРВЕР

На сервер передаются только данные, начинающиеся со знака d, при этом происходит авторизация по логину и паролю. Данные передаются раз в 20 секунд, сервер читает строку по знакам, разбирая ее и передавая данные в

MySQL сервер по отдельности, затем возвращает ответ в виде m0, m1, или m2. С процедурой передачи данных проблем нет, все работало стабильно, ошибок не случилось.

Возвращаемые данные на умную кормушку тоже читались и воспринимались микроконтроллерами без проблем.

4.5. SMOKE ТЕСТИРОВАНИЕ

Тестирование SMOKE предполагает множественное включение и выключение устройства или использование одной функции разными способами и анализ того, как устройство на это отреагирует.

«Умная кормушка» после 100 резких выключений и включений не показала каких-либо ошибок. При этом выключалась она в разных пунктах меню, в разных состояниях датчиков, с записанными и пустыми данными в eeprom.

Проверялась работа с 3.3V и 5V, питанием через датчик WiFi и через Arduino. Все стабильно работало, только lcd дисплей при 3.3V работал медленнее и тусклее.

Тестировалось многократное подключение и отключение зарядки батареи.

Протестирована многократная разрядка и зарядка батареи, а так же удержание зарядки батареи после ее полного заряда (для теста защиты батареи).

Аккумулятор выдерживал без подзарядки от 23-26 часов.

Проблем выявлено не было.

ЗАКЛЮЧЕНИЕ

В представленной работе были разобраны все особенности, связанные с проектированием, разработкой, тестированием и вводом в эксплуатацию «умной» кормушки для домашних животных. Обозначены задачи, которые должны быть решены с использованием данной системы. Создана собственная «умная» кормушка с учетом следующих этапов развития системы: планирования, разработки, внедрения и эксплуатации.

На стадии разработки, внедрения и эксплуатации использовалось свободное ПО: Apache, PHP, Arduino IDE, Android Studio, MySQL. Была дана оценка текущего состояния данной системы, построены функциональные модели с описанием всех рабочих подсистем. Озвучены причины, по которым не производилась некоторые аналоги не могут считаться «умными» кормушками. Так же в ходе анализа системы «умная» кормушка не было выявлено серьезных проблем работы.

По итогам работы, можно сделать следующий вывод:

- получена умная кормушка, работающая по принципам «Интернет вещей»;
- организована серверная, физическая и клиентская части системы «умная» кормушка;
- были предусмотрены экстренные условия работы кормушки;
- теперь можно всегда отслеживать и контролировать питание питомца из любой точки мира.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Гарифуллина А.Р., Крюкова А.А. ТЕХНОЛОГИЯ ИНТЕРНЕТ ВЕЩЕЙ // Международный студенческий научный вестник. – 2017. – № 6. – Дата обращения: март 2020.
- 2 Кота и пса накормит робот – <https://habr.com/ru/company/madrobots/blog/479598/>. – Дата обращения: март 2020.
- 3 Интернет-вещей – <https://habr.com/ru/post/149593/>. – Дата обращения: март 2020.
- 4 Облачные вычисления – <https://habr.com/ru/post/111274/>. – Дата обращения: март 2020.
- 5 Автоматические кормушки для домашних животных – <http://wiki.amperka.ru/slot-box:three-automatic-animal-feeder>. – Дата обращения: март 2020.
- 6 Автоматическая кормушка для кошек и собак Feed-Ex PF7 для сухого корма – <https://zoodialog.ru/katalog/koshki/avtokormushki/avtomaticheskaya-kormushka-dlya-koshek-feed-ex-pf7-dlya-suhogo-korma/>. – Дата обращения: март 2020.
- 7 FEED AND GO. АВТОКОРМУШКА – <https://chipgifts.ru/feed-and-go>. – Дата обращения: март 2020.
- 8 Автоматическая кормушка для животных Furrytail Smart Feeder – <https://mysku.ru/blog/china-stores/77397.html>. – Дата обращения: март 2020.
- 9 Краткий путеводитель по беспроводным технологиям «Интернета вещей». Часть 3. Wi-Fi – <https://ptelectronics.ru/stati/wi-fi-iot-control-engineering/>. – Дата обращения: март 2020.

- 10 Часть 1. Что такое протокол SSL и зачем он нужен? – <https://www.ibm.com/developerworks/ru/library/ac-iscssl1/>. – Дата обращения: апрель 2020.
- 11 Подробное описание интерфейса I2C – <https://radioham.ru/i2c/>. – Дата обращения: апрель 2020.
- 12 Arduino Mega 2560 – <https://arduinoplus.ru/arduino-mega2560/>. – Дата обращения: апрель 2020.
- 13 NodeMcu v2 на ESP8266-12E - <https://robotchip.ru/obzor-platy-nodemcu-v2-na-esp8266-12e/>. – Дата обращения: апрель 2020.
- 14 Oracle MySQL – <https://www.oracle.com/ru/mysql/>. – Дата обращения: апрель 2020.
- 15 Веб-сервер Apache – <http://www.russika.ru/ef.php?s=4005>. – Дата обращения: апрель 2020.
- 16 Язык программирования PHP: от истоков до современности. – https://skillbox.ru/media/code/php_ot_istokov_do_sovremenno_sti/. – Дата обращения: апрель 2020.
- 17 Мобильные приложения для управления всем – <https://iot.ru/gorodskaya-sreda/mobilnye-prilozheniya-dlya-upravleniya-vsem>. – Дата обращения: май 2020.
- 18 Проектирование таблиц для базы данных – <https://oracle-patches.com/common/3280-проектирование-таблиц-для-базы-данных>. – Дата обращения: май 2020.
- 19 Автоматическая кормушка для кота на Arduino – <https://yandex.ru/turbo?text=https%3A%2F%2Fusamodelkina.ru%2F10026-avtomaticheskaya-kormushka-dlya-kota-na-arduino.html>. – Дата обращения: май 2020.