

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

**РАЗРАБОТКА УМНОЙ АВТОМАТИЧЕСКОЙ КОРМУШКИ ДЛЯ
ЖИВОТНЫХ С МОБИЛЬНЫМ ПРИЛОЖЕНИЕМ ДЛЯ
УПРАВЛЕНИЯ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 441 группы

направления 02.03.03 Математическое обеспечение и администрирование
информационных систем

факультета компьютерных наук и информационных технологий

Талалайкиной Елизаветы Игоревны

Научный руководитель:

зав. кафедрой, к.ф.-м.н., доцент.

М.В. Огнева

подпись, дата

Зав. кафедрой:

к.ф.-м.н., доцент

М.В. Огнева

подпись, дата

Саратов 2020

ВВЕДЕНИЕ

Разнообразие и возможности современных технологий поражают воображение и превосходят самые смелые ожидания и предсказания ученых прошлых веков. Одной из самых интересных и быстроразвивающихся технологий является так называемый Интернет вещей.

Интернет вещей (Internet of things, IoT) – это технология, объединяющая устройства в единую сеть и позволяющая им проводить сбор, анализ, обработку и передачу данных другим устройствам при помощи программного обеспечения. Устройства работают в режиме реального времени как без участия человека, так и с некоторым контролем пользователя, например, настройкой, выдачей инструкций или предоставлением доступа к данным. Внедрение Интернета вещей произошло благодаря широкому распространению интернета, смартфонов и беспроводных сетей.

Сейчас устройства из категории Интернета вещей можно найти практически для любой составляющей повседневной жизни человека, в том числе и в товарах для животных.

Питомец – это не только большая радость, но и большая ответственность. В современном динамическом мире невозможно точно знать, когда и насколько может понадобиться уехать, и не во все поездки возможно забрать питомца. В таких условиях появление на рынке автоматических кормушек для животных стало хорошим решением для людей, которым приходится часто бывать в рабочих поездках или задерживаться допоздна на работе.

Изначально такие устройства представляли собой бак, в который насыпался сухой корм, выдающийся с установленным интервалом в миску питомца. Позже автоматические кормушки стали усложняться, например, добавились возможности контролировать размер порций или гибко настраивать периодичность выдачи порции. Сейчас на рынке представлены модели, управление которыми возможно не только вручную, но и удаленно, через мобильные приложения. Такие кормушки могут поддерживать запись

голосовых сообщений и их воспроизведение на устройстве, или даже двустороннюю аудиосвязь с питомцем, видеонаблюдение через установленную на кормушке камеру, задание расписания кормления на смартфоне, уведомления об успешной выдаче корма или возникающих проблемах в работе, автономную работу при отсутствии соединения с интернетом или электричества.

Интересными и новыми функциями приложений для автоматических кормушек являются сбор и анализ статистики по съеданному питомцем корму. Если питомец изменяет пищевое поведение (ест меньше или больше нормы), это может быть сигналом развития заболевания. Приложение, отслеживающее такие изменения, помогает как можно быстрее заметить нетипичное поведение питомца, обратиться к ветеринару и начать своевременное лечение.

Таких кормушек на данный момент у крупных и известных производителей умных товаров для животных нет. Есть пример стартапа по производству такой кормушки, но, судя по отзывам пользователей, с функцией сбора данных это устройство справляется очень плохо.

Цель бакалаврской работы – разработка умной автоматической кормушки для кошек и маленьких собак с функцией сбора, хранения и предоставления пользователю статистики по съеданному питомцем корму и мобильного приложения для управления данным устройством.

Поставленная цель определила **следующие задачи**:

1. изучить общую информацию об Интернете вещей;
2. изучить представленные на рынке варианты кормушек, выделить их достоинства, недостатки и особенности;
3. продумать конструкцию кормушки, подобрать подходящий механизм выдачи корма, определиться с электронной частью конструкции;
4. изучить электронные компоненты, используемые в конфигурации кормушки, и документацию к ним;
5. продумать конфигурацию приложения для управления кормушкой, его серверную и клиентскую части;

6. собрать кормушку;
7. разработать серверную и клиентскую части приложения;
8. протестировать работу сделанного умного устройства.

Методологические основы разработки приложения для управления кормушкой представлены в работах Марка Лутца, Мигеля Гринберга, Джозефа Хокинга, Алана Торна, Стивена Гаосэ.

В теоретической части изложена общая информация об Интернете вещей, рассмотрены представленные на рынке варианты кормушек, выделены их достоинства, недостатки и особенности, описаны конструкция кормушки, механизм выдачи корма и электронная часть конструкции, представлена информация об электронных компонентах, использованных в конфигурации кормушки, описана конфигурация приложения для управления кормушкой, его серверная и клиентская части, рассмотрены средства разработки для создания серверной и клиентской частей приложения.

В практической части описан процесс сбора конструкции кормушки и ее электронной части, рассмотрен процесс разработки серверной и клиентской частей приложения.

Структура и объём работы. Бакалаврская работа состоит из введения, пяти разделов, заключения, списка использованных источников и двух приложений. Общий объем работы – 175 страниц, из них 121 страница – основное содержание, включая 26 рисунков, список использованных источников информации – 48 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Общие сведения об области разработки» посвящен рассмотрению в общих чертах технологии «Интернет вещей», истории ее появления и развития, основных рисков в данной сфере, а также обзору существующих на рынке автоматических кормушек, их достоинств и недостатков.

Интернет вещей (Internet of things, IoT) – это технология, объединяющая устройства в единую сеть и позволяющая им проводить сбор, анализ, обработку и передачу данных другим устройствам при помощи программного обеспечения.

IoT-системы состоят из сети устройств и интернет-платформы, к которой они подключены разными способами (WiFi, Bluetooth, сотовая связь). Устройства собирают данные, которые затем отправляются в облако. После попадания данных в облако система обрабатывает их и выполняет необходимые действия (оповещение пользователя, включение или выключение какого-то устройства, изменение параметров его работы).

Концепция Интернета Вещей появилась в 1982 году. В 1999 году Кевином Эштоном впервые было употреблено словосочетание «Интернет вещей» (Internet of Things, IoT).

В 1990-х вся деятельность, связанная с IoT, носила в основном теоретический характер – концепции, обсуждения, отдельные идеи, а в 2000-х и 2010-х стали массово появляться и запускаться успешные IoT-проекты в реальности. Было разработано множество пользовательских устройств, относящихся к Интернету вещей. Кроме того, начали развиваться масштабные проекты, основанные на технологиях IoT – умные города, умное производство, умный транспорт, беспилотные автомобили и многое другое.

Чем больше функций может выполнять умное устройство, тем больше данных об окружающей среде и пользователе ему необходимо собрать. Проблема безопасности и конфиденциальности этих персональных данных

остро стоит в технологии «Интернет вещей», и способами ее решения являются сертификация и блокчейн.

В данном разделе было рассмотрено 15 автоматических кормушек от разных производителей. Базовым функционалом автоматических кормушек с приложением для смартфона является выдача корма по задаваемому в приложении расписанию и немедленная выдача порции при выборе соответствующего пункта в приложении. Некоторые автоматические кормушки могут поддерживать запись голосовых сообщений и их воспроизведение на устройстве, или даже двустороннюю аудиосвязь с питомцем, видеонаблюдение через установленную на кормушке камеру и управление с помощью голосовых помощников.

Рассмотренные автоматические кормушки крупных производителей не имеют функции работы со статистикой по съеданному питомцем корму, которая реализована у кормушки, созданной в рамках данной бакалаврской работы.

Второй раздел «Теоретические сведения о конфигурации кормушки» посвящен описанию конструкции кормушки, созданной в рамках данной бакалаврской работы, и рассмотрению её электронных компонентов.

Данная кормушка является кормушкой с бункером, корм из которого подается с помощью так называемого «винта Архимеда», или шнека, в миску, снизу которой закреплен тензодатчик, позволяющий снимать показание о весе корма, находящегося на миске. Все электронные компоненты кормушки находятся под бункером для корма.

Для создания электронной части кормушки и организации связи с приложением на телефоне понадобился ряд электронных компонентов:

1. одноплатный микрокомпьютер Orange Pi One – основной управляющий модуль кормушки, подающий сигналы на остальные электронные компоненты и принимающий информацию с них. Содержит серверную часть приложения и обрабатывает информацию о состоянии кормушки;

2. реверсивный широтно-импульсный преобразователь постоянного тока (реверсивный ШИМ) отвечает за регулицию скорости вращения моторчика и переключение направления движения;
3. инфракрасный датчик позволяет измерять скорость вращения мотора;
4. мотор-редуктор соединяется непосредственно со шнеком, обеспечивая его вращение при подаче сигнала;
5. тензодатчик и модуль HX711 – связка, которая позволяет измерять давление, оказываемое на поверхность датчика, по полученным данным вычислять вес, и передавать показания на микрокомпьютер.

Третий раздел «Теоретические сведения об используемых в разработке приложения для кормушки технологиях» посвящен описанию конфигурации разработанного в рамках данной бакалаврской работы приложения и рассмотрению использованных при разработке технологий.

Архитектура разработанного приложения представляет собой модель клиент-сервер. Клиентом является приложение на смартфоне, написанное в межплатформенной среде разработки Unity, а сервером – находящаяся на микрокомпьютере Orange Pi One и написанная на Python часть приложения. Клиентская часть позволяет настраивать расписание выдачи корма и размер порций, получает и хранит статистику по съеданному ежедневно корму, позволяет выдавать порции заданного веса вне расписания. Серверная часть приложения отвечает за работу с электронными компонентами кормушки. Также на сервере в формате JSON хранится расписание кормления. Благодаря этому даже при возможных неполадках в соединении с сервером корм будет выдаваться по сохраненному расписанию.

Для создания серверной части приложения были использованы микрофреймворк Flask и библиотека ruA20.

Flask позволяет полностью контролировать приложения разработчиком и с легкостью расширять функционал. Он включает в себя базовую функциональность, необходимую всем веб-приложениям, а все остальное

может быть предоставлено сторонними расширениями и самим программистом.

Библиотека `руA20` упрощает работу с пинами микрокомпьютера, то есть при помощи нее можно по команде на каждый из пинов подавать сигнал, а также считывать информацию с них.

Клиентская часть представляет собой приложение на смартфоне, написанное в межплатформенной среде разработки Unity. Unity, в первую очередь, является игровым движком, но данная среда разработки удобна и для создания прикладных приложений.

Причин для выбора Unity как платформы для разработки несколько:

1. в Unity удобно работать с пользовательским интерфейсом приложения. Все элементы пользовательского интерфейса можно установить на нужное место, выставить им требуемые свойства и привязать к ним необходимые скрипты;
2. в данном приложении необходима связь с сервером, и Unity предоставляет такую возможность. С помощью стандартного класса `UnityWebRequest` можно легко составлять HTTP-запросы и обрабатывать HTTP-ответы;
3. в клиентской части приложения необходима сериализация данных, например, информации о питомце и статистике кормления. Unity предоставляет разработчику встроенный механизм сериализации данных в разных форматах.

Четвертый раздел «Сборка устройства кормушки» посвящен описанию процесс сбора конструкции кормушки и работе с ее электронной частью (используемые элементы питания, типы соединительных проводов, процесс настройки микрокомпьютера Orange Pi One и процесс соединения электронных компонентов между собой).

Конструкция кормушки была собрана из пластиковых контейнеров. Шнек был собран из кругов, вырезанных из пластикового контейнера, закрепленных на ПВХ трубе диаметром 16мм. Шнек помещен в ПП трубу диаметром 50мм (труба для подачи корма).

Микрокомпьютер Orange Pi One настроен по инструкции с официального сайта производителя. К микрокомпьютеру подключены все компоненты, составляющие электронную часть кормушки.

Пятый раздел «Разработка приложения» посвящен описанию скриптов, используемых в серверной и клиентской частях разработанного приложения.

Скрипт серверной части приложения, написанный на языке программирования Python, находится на микрокомпьютере. Он отвечает за взаимодействие с электронными компонентами, принятие и отправку HTTP-запросов и сохранение расписания кормления.

Клиентская часть приложения разработана в мультиплатформенной среде разработки Unity и представлена приложением для операционной системы iOS.

Приложение состоит из четырех вкладок: «Основное», «Расписание кормления», «Статистика» и «Настройки».

На вкладке «Основное» отображается информация о питомце, которую можно отредактировать на вкладке «Настройки», расписание кормления на день, которое можно отредактировать на вкладке «Расписание кормления», в котором отмечаются выданные порции. Кроме того, на данной вкладке можно выдать порцию необходимого размера вручную.

На вкладке «Расписание кормления» можно отредактировать количество корма за день и расписание кормления. Максимально в расписании кормления допускается двенадцать порций.

На вкладке «Статистика» можно получить информацию о питании питомца. В виде графиков и текста можно вывести данные на выбранную дату о том, сколько корма было выдано по расписанию, сколько – вне расписания, через функцию выдачи корма в приложении, и сколько из этого количества питомец не съел.

На вкладке «Настройки» можно отредактировать информацию о питомце, которая будет сохранена и отображена на вкладке «Основное».

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной выпускной квалификационной работы была изучена общая информация об Интернете вещей, история появления и развития технологии и основные проблемы в данной области. Были изучены представленные на рынке варианты кормушек, выделены их особенности, достоинства и недостатки.

Была создана конструкции электронной кормушки с подобранными изученными электронными компонентами и подходящим механизмом выдачи корма.

Для создания приложения были продуманы его серверная и клиентская части. Было создано приложение для управления кормушкой для мобильной платформы iOS. При необходимости данное приложение может быть перенесено также на мобильную платформу Android.

Созданное приложение имеет следующие функции: создание расписания кормления и автоматическая выдача порций согласно этому расписанию, выдача порций задаваемого пользователем веса вручную, сбор и показ статистики по выданному и не съеденному питомцем корму, хранение информации о питомце (имя, возраст, вес, пол).

В ходе проверки работоспособности созданного умного устройства на протяжении трех недель была выявлена его стабильная работа (корм выдается по заданному расписанию и по команде пользователя, информация о статистике передается с сервера на приложение на стороне клиента).

В продолжение разработки можно перенести электронную часть в другой корпус (не самодельный, а напечатанный при помощи 3D-печати), и добавить дополнительные функции созданному умному устройству, например, видеонаблюдение за питомцем и мониторинг количества корма в баке.

Таким образом, все поставленные задачи были выполнены, цель работы достигнута.

Основные источники информации:

1. Марк Лутц. Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с., ил.
2. Grinberg M. Flask Web Development. Developing web applications with Python – O'Reilly Media, 2014 – 258 p.
3. Джозеф Хокинг. Unity – в действии. Мультиплатформенная разработка на C# : [рус.]. – 2. – СПб : Питер, 2016.
4. Алан Торн. Искусство создания сценариев в Unity : [рус.]. – СПб : ДМК, 2016.
5. Stephen Gose. Making Multiplayer Online Games: A Game Development Workbook for any Phaser JavaScript Gaming Framework. Scribl, 7 нояб. 2016 г. – 253 p.