

**RASPBERRY PI DISPENSADOR DE COMIDA PARA MASCOTAS**

**TESINA**

QUE PARA OBTENER EL GRADO DE

**ING. SISTEMAS DIGITALES INTERACTIVOS**

PRESENTA

**DANIEL EDUARDO GASTÉLUM ÁLVAREZ**

**LUIS FERNANDO COTA RODRÍGUEZ**

ASESOR DE METODOLOGÍA

**M.C. LISBIA ARELI PEREZ ZAMUDIO**

LOS MOCHIS, SINALOA. FEBRERO 2018

**DEDICATORIAS**

Les dedicamos esta tesina principalmente a nuestros seres queridos que nos han apoyado en el transcurso de nuestra estadía en la universidad.

Agradecemos a los maestros que nos apoyaron con su enseñanza y conocimiento de los distintos procedimientos que utilizamos tanto en la tesina, como en la elaboración del dispensador en sí.

Agradecemos a el personal de Tecnologías de la Información (TI) que nos ha apoyado en la obtención de material y la elaboración del prototipo.

A nuestros compañeros de clase por habernos apoyado mutuamente en la elaboración de los apartados de la tesina.

A las mascotas que principalmente serán los más beneficiados al terminar la elaboración del dispensador de alimentos.

**RESUMEN**

Para esta tesina se escogió el tema de un dispensador de comida para mascotas utilizando la tecnología de Raspberry pi por la necesidad de implementar la tecnología en beneficio a nuestras mascotas. El problema surge de la necesidad de proporcionales alimento a nuestras mascotas y que por inconvenientes ajenos se nos puede dificultar esta simple tarea, ya sea por salir de viaje, algún descuido u olvido de darle alimento a la mascota, esta tarea se simplificará con la implementación del dispositivo. Al proponer un sistema enfocado a la domótica surgen variantes de elaboración que dentro del marco metodológico se presentan y se explican. Dentro de estas variantes se presentaron propuestas interesantes para hacer del dispensador un dispositivo que cubra más necesidades de la mascota.

Para la realización de marco metodológico se defino la investigación como experimental en su categoría: experimento puro. Para contemplar la población se determino es los hogares de la ciudad de Los Mochis, Sinaloa; en la muestra de probabilidad. El uso de métodos de investigación se determinó como empíricos en la categoría método experimental usando los procedimientos análisis y abstracción.

Para obtener los resultados de la investigación se utilizó el instrumento de encuesta y los resultados obtenidos de esta investigación se encontraran al final de esta tesina, en el apartado de anexos en forma de gráficas.

El elaborar un dispositivo con tal giro ayudara a muchos hogares pero sobre todo a las mascotas por el apoyo que este sistema les brindara.

INDICE

|  |  |  |
| --- | --- | --- |
| **CAPÍTULO I. PLANTEAMIENTO DEL PROBLEMA** | |  |
| 1.1 Justificación del estudio y antecedentes del problema   * Descripción de la situación problemática | 6 | | |
| * 1. Definición del problema  1. Pregunta general 2. Preguntas de investigación | 8 | | |
| * 1. Objetivos de investigación  1. General 2. Específicos | 9 | | |
| * 1. Hipótesis | 10 | | |
| 1.5 Delimitación del estudio | 11 | | |
| **CAPÍTULO II. MARCO TEÓRICO** |  | | |
| 2.1. ¿Qué es un dispensador inalámbrica de comida para mascotas? | 12 | | |
| 2.1.1. ¿Qué es Raspberry Pi? | 13 | | |
| 2.1.1.1 Sistemas Operativos de Raspberry Pi | 14 | | |
| 2.1.2. NodeJs | 18 | | |
| 2.1.2.1 Node-RED | 22 | | |
| 2.1.3. Características del dispensador | 23 | | |
| 2.1.3.1 Servidores para el dispositivo | 24 | | |
| 2.2 Creación del prototipo | 28 | | |
| 2.2.1 Materiales para su elaboración | 29 | | |
| 2.2.2 Pasos a seguir para su construcción | 30 | | |
| 2.2.3 Programación adecuada | 32 | | |
| 2.3. Funcionamiento del prototipo | 36 | | |
| 2.3.1 Programación de tareas (Cron) | 37 | | |
| 2.4 Diferentes ejemplos de la idea | 40 | | |
| 2.5 Beneficios de su implementación | 42 | | |
| 2.6 Costos y tiempo | 43 | | |
| 2.7 Adaptaciones futuras | 44 | | |
| **CAPÍTULO III. MARCO METODOLÓGICO** |  | | |
| 3.1 Definición del tipo de investigación a realizar | 46 | | |
| 3.2 Población y muestra | 47 | | |
| 3.3 Métodos de investigación | 48 | | |
| 3.4 Operacionalización de las variables | 49 | | |
| 3.5 Instrumentos de medición para recolección de datos | 51 | | |
| 3.6 Estudios piloto | 51 | | |
| 3.7 Procedimientos para el análisis de datos | 51 | | |
| **CAPÍTULO IV. ANÁLISIS DE LOS RESULTADOS Y PROPUESTA** |  | | |
| 4.1 Análisis de los resultados | 52 | | |
| 4.2 Propuesta de la investigación | 55 | | |
| **CONCLUSIONES** | 56 | | |
| **RECOMENDACIONES** | 58 | | |
| **GLOSARIO DE TÉRMINOS** |  | | |
| **BIBLIOGRAFÍA** | 59 | | |
| **ANEXOS** | 63 | | |

ÍNDICE DE TABLAS

Tabla 1. Tabla de control 37

Tabla 2. Tabla de costos 42

Tabla 3. Tabla de variables y operaciones 50

Tabla 4. Tabla de costos 53

**CAPÍTULO I. PLANTEAMIENTO DEL PROBLEMA**

**1.1 Antecedentes del problema.**

¿Qué es un dispensador de comida para mascotas Inalámbrico? El interés por hacer un dispensador de comida para mascotas surge la problemática de que el animal se quede solo en el hogar durante un viaje, ya sea personal o de negocios, y la opción de dejar lleno el plato de comida de la mascota cuando se está fuera de casa deja de ser viable puesto que se desperdicia mucha comida que el animal no se come. Con la idea planteada en esta parte de la tesina se resolverán los siguientes cuestionamientos: ¿Qué es un dispensador de comida para mascotas?, ¿Qué es Raspberry Pi?, ¿Cómo funciona dicho aparato?, ¿Qué materiales se necesitan?, ¿Cuáles son los beneficios de un dispensador de comida para mascotas?, ¿Cuáles son los costos que implica la implementación del sistema?, ¿Podría repercutir en la vida del animal? Al responder todos estos cuestionamientos se obtendrá información específica para la realización del proyecto.

David Bryan publicó en su blog de internet en el año 2013 sus antecedentes de sus proyectos realizados con su Rapsberry Pi, la cual no podía quitársela de la cabeza, su problemática empieza cuando él y su novia programan un viaje de fin de semana por el cual no podrían alimentar a sus 2 gatos mascota. El luego explica que decide solucionar el problema construyendo un dispensador de comida remoto. En su blog empieza a describir paso a paso como lo realizo incluyendo el material y su procedimiento. Basándonos en su historia y debido a que las personas de la ciudad de los Mochis, Sin. Tienen la misma problemática se ha optado por realizar el experimento y ponerlo a prueba.

El sitio donde se realizara la investigación del proyecto será en la ciudad de Los Mochis, Sin. México, específicamente en hogares donde halla mascotas. Se tendrá el apoyo de la Institución Universidad del Valle del Fuerte, el departamento de Tecnologías de la Información de esa misma institución y personal docente como asesor. El proyecto de investigación involucrará datos de fuentes de información de la actualidad para solventar la problemática estimando un tiempo de realización de 1 año. Según en la página de David Bryan (*drstrangelove),* en su artículo de esta misma página. (Bryan D. 2013), muestra las herramientas necesarias que a continuación se indicaran:

Herramientas necesarias: soldador, taladro, broca de ½ ", broca de 1" (para interruptor de botón opcional), pistola de pegamento caliente + pegamento, cortador de alambre, alicates.

Requisitos previos para la construcción: tenga todas las piezas a mano, consulte la hoja de cálculo de Google. Tenga un sistema operativo que funcione y esté instalado en el Pi. AdafruitOccidentalis o Raspbian funcionan bien. Tener la configuración del Adaptador Wifi lista para usar: Occidentalis o Raspbian.

Juega con el GPIO dentro y fuera del Pi descargando el código de mi Repositorio de GitHub.

Instalar el módulo Python GPIO.PWM: RPIO install de los documentos.

El departamento de TI de UNIVAFU brindara el apoyo en algunas cuestiones materiales como la Raspberry Pi, algunos servomotores, cableado del sistema así como los docentes apoyarán en la cuestión de la programación y testeo del proyecto.

La solución que emplea el proyecto es que por medio de la conexión de internet, estando fuera del hogar, mandar una señal por medio de este y que el dispositivo Raspberry estando conectado a la conexión WiFi del hogar, reciba la señal, mueva un servomotor que permita el paso del alimento y lo deposite en el plato de comida de la mascota en cuestión haciendo más fácil la manera de poder alimentar al animal.

Este proyecto resolvería los problemas de personas con constantes viajes tanto de negocios como personales y que no pudiesen llevar consigo a la mascota y por ende dejarla sola en casa

Varios estudios de esta problemática ya han sido puestos en investigaciones y todos en general buscan la misma solución, alimentar a la mascota cuando se está fuera de casa. La innovación de nuestro proyecto implica hacerlo mediante conexiones inalámbricas por medio tanto de un dispositivo móvil o de un ordenador por medio de páginas web.

Las posibles causas y consecuencias de realizar el experimento son el poner a prueba el dispensador y ver si la problemática se pudo resolver la cual es si el proyecto funcionara y si la macota será alimentada de lo contrario se tendrá que recurrir a métodos más convencionales como pedirle a alguien que alimente a la mascota o dejar un enorme plato de comida y agua suficiente hasta el regreso del dueño a la casa.

**1.2 Justificación**

Los motivos y el interés por la cual este proyecto es desarrollado surgen de necesidades del día a día como alimentar a una mascota del hogar, en algunos casos se vuelve algo tedioso sobre todo si hay muchos periodos de alimentación de la mascota, esto es una problemática en muchos hogares ya que no hay manera de realizar un seguimiento de la alimentación adecuada de las mascotas, es por eso que este prototipo ayudara a solventar esta problemática. Al investigar este problema los principales beneficiaros serán los dueños de las mascotas que tendrán un mejor seguimiento a la hora de alimentar a su mascota y las mismas mascotas que tendrán su alimento cuando sea necesario y las horas acordadas.

Las aportaciones o el impacto social que se obtendrá es la mejora en cuanto a la organización de tiempos que el animal tendrá para tener sus alimentos que le corresponden (dependiendo de cada dueño).

Es cierto que la mayoría de la sociedad aquí en la ciudad de Los Mochis, sino es que todos, tenemos una mascota que cuidar y hay que decir que un tanto porcentaje se preocupa cómo debe de ser por la alimentación de su mascota, es por eso que este sistema servirá como complemento en esta cuestión.

En base a las encuestas realizadas dieron los siguientes resultados estadísticos 90% del entrevistado tienen mascotas y de este 90% un 80% se le complica dejar sola a su mascota 90% de los entrevistados les gusta la idea y 60% pagarían el precio.

1.3 **Objetivo de investigación**

Objetivo general:

* Diseñar e implementar un sistema dispensador de comida para mascotas que funcione inalámbricamente por medio de la conexión a internet utilizando los recursos de una microcomputadora Raspberry Pi.

Objetivos específicos:

* Conocer los beneficios que proporcionaría la implementación de dicho sistema en los hogares.
* Averiguar los materiales, costos y tiempo de implementación del prototipo.
* Comparar el sistema con otros proyectos ya desarrollados.
* Describir el funcionamiento de dispensadores controlados remotamente.
* Verificar si es factible implementar dispensadores remotos en los hogares.
* Observar el impacto de la implementación en los hogares de la cuidad de Los Mochis.

**1.4 Pregunta Central**

Pregunta Central:

• ¿Cómo diseñar e implementar un sistema dispensador de comida para mascotas que funcione inalámbricamente por medio de la conexión a internet utilizando los recursos de una microcomputadora Raspberry Pi?

**1.5 Preguntas complementarias**

Preguntas complementarias:

• ¿Qué beneficios tiene poder alimentar a su mascota de forma remota?

• ¿Cuáles son los materiales que se necesitan para crear dicho prototipo?

• ¿Cuál sería el impacto de la implementación de dispensadores de comida remotos en la ciudad de los Mochis, Sinaloa?

• ¿Para qué el uso de dispensadores controlados remotamente?

• ¿Es posible poder implementar los dispensadores en todas las casas de las personas?

• ¿Cuánto tiempo tomaría diseñar, implementar y evaluar el prototipo?

• ¿Qué diferencia tiene nuestro dispensador de alimentos con otros ya existentes?

**1.6 Hipótesis o Supuestos**

"El dispensador de alimentos automático beneficia a los dueños de las mascotas al brindar el alimento sin necesidad de dejar el plato con comida extra o buscar a otra persona de confianza que alimente a dicha mascota."

**Delimitación del estudio**

En la elaboración de esta investigación se obtendrá ayuda del departamento de TI de la Universidad del Valle del Fuerte y personal docente de la misma. La investigación contemplara solo la elaboración del prototipo y su implementación en los hogares de la cuidad de Los Mochis, Sin. México. El proceso de la investigación se realizará en un rango de Agosto-Diciembre del año 2015. Una complicación al realizar la investigación sería el tiempo, ya que los materiales se pedirán en diferente sitios web y el tiempo de entrega de dichos materiales podría retrasar el tiempo de ejecución del proyecto.

**CAPÍTULO II. MARCO TEÓRICO**

**2.1 ¿Qué es un dispensador inalámbrico de comida para mascotas?**

Para entender mejor el concepto de que es un dispensador inalámbrico de comida para mascotas, veremos qué significado tiene cada concepto. El verbo dispensar quiere decir: Dar, repartir, distribuir o suministrar algo(WordReference, 2005), entonces podemos inferir que un dispensador es un dispositivo que suministra un material. La palabra inalámbrico se refiere que no utiliza cables, es decir, cualquier tecnología que permite una comunicación entre dispositivos sin ninguna conexión física visible (Alegasa, 2010).Ahora bien, un dispensador inalámbrico de comida para mascotas es aquel que suministra alimento para dicho animal de manera remota, es decir, sin conexión física visible. En esta ocasión para lograr una comunicación con el aparato de manera remota se utilizara un dispositivo llamado Raspberry Pi que se explicará a continuación.

**2.1.1. ¿Qué es Raspberry Pi?**

La Raspberry Pi es un ordenador del tamaño de una tarjeta de crédito que se conecta a un monitor, un teclado y un mouse estándar. Es un dispositivo pequeño que permite a las personas de todas las edades explorar un poco más la computación y aprender a programar en lenguajes como Scratch y Python (Raspberry Pi Fundación, 2015).

Raspberry Pi  usa mayoritariamente sistemas operativos basados en el núcleo Linux. Raspbian, una distribución derivada de Debian que está optimizada para el hardware de Raspberry Pi, se lanzó durante julio de 2012 y es la distribución recomendada por la fundación para iniciarse. Otros sistemas operativos que son compatibles son: AROS, Android, Arch Linux ARM, Debian, Firefox OS, Gentoo Linux, Google Chromium OS, Kali Linux, Open webOS, PiBang Linux, Pidora, QtonPi, Raspbian, Slackware ARM, Unix, RISC OS 5, Windows 10 (Wikipedia, 2015).

**2.1.1. Sistemas operativos de Raspberry Pi**

**Raspbian**

Raspbian es un sistema operativo libre basado en Debian optimizado para el hardware Raspberry Pi. Un sistema operativo es el conjunto de programas básicos y utilidades que hacen que su funcionamiento Raspberry Pi. Sin embargo, Raspbian ofrece más que un SO puro; viene con más de 35.000 paquetes, software pre-compilado liado en un formato que hace más fácil la instalación en su Raspberry Pi.

La construcción inicial de más de 35.000 paquetes de Raspbian, optimizado para un mejor rendimiento en el Raspberry Pi, se terminó en junio de 2012. Sin embargo, Raspbian está todavía en desarrollo activo con un énfasis en la mejora de la estabilidad y el rendimiento de la mayor cantidad de paquetes de Debian como sea posible.

Nota: Raspbian no está afiliada con la Fundación Raspberry Pi. Raspbian fue creado por un pequeño dedicado equipo de desarrolladores que son fans del hardware Raspberry Pi, los objetivos educativos de la Fundación Raspberry Pi, y, por supuesto, el proyecto Debian (Raspbian, 2015).

Raspbian es un puerto no oficial de Debian Wheezy arm hf con ajustes de compilación ajustados para producir código optimizado "flotar duro" que se ejecutará en la Raspberry Pi. Esto proporciona un rendimiento significativamente más rápido para aplicaciones que hacen un uso intensivo de las operaciones de aritmética de punto flotante. Todas las otras aplicaciones también ganarán algo de rendimiento a través del uso de las instrucciones avanzadas de la CPU ARMv6 en Raspberry Pi.

Aunque Raspbian es principalmente a los esfuerzos de Mike Thompson (mpthompson) y Peter Green (plugwash), también se ha beneficiado enormemente del apoyo entusiasta de miembros de la comunidad Raspberry Pi que deseen obtener el máximo rendimiento de su dispositivo. (Raspbian, 2015)

**Ubuntu Mate**

Ubuntu es uno de, si no el más grande, desplegados Linux los sistemas operativos de escritorio basados ​​en el mundo. Linux es el núcleo de Ubuntu y hace posible la creación de sistemas operativos seguros, potentes y versátiles, como Ubuntu y Android. Android está ahora en manos de miles de millones de personas en todo el mundo y también es alimentado por Linux.

Ubuntu está disponible en un número de diferentes sabores, cada uno viene con su propio entorno de escritorio. Ubuntu MATE lleva el sistema operativo base Ubuntu y añade el escritorio MATE.

El escritorio MATE es un tal implementación de un entorno de escritorio e incluye un gestor de archivos que puede conectarse a los archivos locales y en red, un editor de texto, calculadora, gestor de archivos, visor de imágenes, visor de documentos, monitor del sistema y terminales. Todos los cuales son altamente personalizable y gestionado a través de un centro de control.

MATE Desktop proporciona un entorno de escritorio intuitivo y atractivo el uso de metáforas tradicionales, lo que significa que si alguna vez has usado Microsoft Windows o Apple Mac OS, que se siente muy familiar.

El MATE de escritorio tiene una rica historia y es la continuación del escritorio GNOME2, que era el entorno de escritorio por defecto en muchos sistemas operativos Linux y Unix desde hace más de una década. Esto significa que MATE Desktop se trató, probado y muy fiable (Ubunt Mate, 2015)

**Windows 10 IoT**

Windows 10 IoT Core es una versión muy limitada de Windows 10 que no cuenta con muchas de las opciones de su hermano mayor porque, simplemente, no las necesita. Para empezar, se trata de una edición para arquitecturas de 32 bits y concretamente para la ARMv7. No contaremos con un entorno de escritorio: nada de menú de inicio, nada de Cortana, y nada de ejecutar aplicaciones o juegos tradicionales de Windows 10 que queramos ver en nuestro monitor.

No. En Windows 10 IoT Core nos encontramos con una plataforma hecha por desarrolladores y para desarrolladores. Los responsables de Microsoft lo dejaban claro en el anuncio público de esta edición final, y además de hablarnos de las mejoras desde aquellas primeras versiones preliminares explicaban cuáles eran los posibles problemas con los que se podrían encontrar los usuarios en esta edición.

Es algo que cualquier usuario de las Raspberry Pi 2 tiene que tener claro: Windows 10 IoT Core no es una plataforma que nos vaya a permitir abrir un navegador ligero, editar documentos de Office en nuestro MiniPC o escuchar música de buenas a primeras. La idea con esta plataforma es la de invitar a los desarrolladores a crear e inventar todo tipo de soluciones adaptadas a la Internet de las Cosas y no centradas en el segmento de los PCs y portátiles tradicionales.

Windows 10 IoT Core está dirigido claramente a la generación de makers que a menudo nos sorprenden con sus desarrollos en Arduino, RPi y otras plataformas de este tipo. Y eso se nota tanto en el trabajo y disfrute con la plataforma como en la propia instalación del sistema operativo en estos pequeños miniPCs. (Pastor J., 2015)

**RetroPie**

El realismo en los videojuegos es uno de los objetivos que las grandes compañías, dedicadas a este mundillo, llevan años persiguiendo y que con el paso de los años y los avances en tecnología, se va consiguiendo en cierta manera. Todavía falta mucho hasta que consigamos ponernos en la piel de Cristiano Ronaldo y ejecutar sus movimientos con nuestro propio cuerpo, pero al menos su aspecto virtual en el FIFA va mejorando cada año y su cara está menos pixelada.

Hasta que esto pase, si superas el cuarto de siglo, eras fan incondicional de las plataformas en 2D y tienes una Raspberry Pi, esta entrada te va a gustar. Vamos a convertir nuestra RPi en una máquina recreativa capaz de ejecutar todos aquellos juegos usando RetroPie. (Benítez M., 2015)

**2.1.2 NodeJs**

Node es un intérprete JavaScript del lado del servidor que cambia la noción de cómo debería trabajar un servidor. Su meta es permitir a un programador construir aplicaciones altamente escalables y escribir código que maneje decenas de miles de conexiones simultáneas en una sólo una máquina física.

Node.js es una librería y entorno de ejecución de E/S dirigida por eventos y por lo tanto asíncrona que se ejecuta sobre el intérprete de JavaScript creado por Google V8.

Operación

Node.js es un entorno JavaScript del lado del servidor, basado en eventos. Node ejecuta JavaScript utilizando el motor V8, desarrollado por Google para uso de su navegador Chrome. Aprovechando el motor V8 permite a Node proporciona un entorno de ejecución del lado del servidor que compila y ejecuta JavaScript a velocidades increíbles. El aumento de velocidad es importante debido a que V8 compila JavaScript en código de máquina nativo, en lugar de interpretarlo o ejecutarlo como bytecode. Node es de código abierto, y se ejecuta en Mac OS X, Windows y Linux. (Smirnov A, 2015)

Ventajas:

En primer lugar, para el rendimiento y la escalabilidad. Node es rápido

Node es también perfecta para ofrecer una API REST – un servicio web que tiene unos parámetros de entrada y pasa un poco de datos de nuevo – la manipulación de datos sencillo y sin una gran cantidad de cálculos.

Desventajas

Como la mayoría de las nuevas tecnologías, que no es tan fácil de implementar Node en los hosts existentes.

Otra desventaja importante de Node es que está todavía en las primeras etapas de desarrollo, lo que significa algunas características pueden cambiar según avanza el desarrollo.

Otras desventajas que se mencionan a menudo (pero que están solucionándose día a día son:

API Inestable: La API de Node tiene la mala costumbre de cambiar en formas que rompen la compatibilidad hacia atrás de versión en versión, lo que requiere que apliques cambios frecuentes en tu código para mantener todo funcionando en las versiones más actuales.

Falta de una Librería Estándar: JavaScript es unos lenguajes con un buen núcleo pero con una flaca librería estándar. Cosas que darías por hecho en otro lenguaje del lado del servidor simplemente no existen.

Falta de Librerías en General: ¿Necesitas una interfaz de bases de datos madura? ¿Un ORM? ¿Una librería de procesamiento de imágenes? ¿Un analizador XML? Como JavaScript no sido popular en el lado del servidor todo esto es muy reciente, o no está probado o está en camino.

Muchas Formas de Programar: La falta inherente de organización de código se puede considerar una gran desventaja. Se nota su efecto claramente cuando el equipo de desarrollo no está muy familiarizado con la programación asíncrona o los patrones de diseño estándar. Simplemente hay demasiadas formas de programar y de obtener código desparejo y difícil de mantener.

No está Probado lo Suficiente. Este punto puede ser susceptible a opiniones subjetivas debido a que es una cuestión bastante abierta. Mientras no tengamos grandes proyectos en producción por varios años, no podremos saber dónde está el problema. (Solo Electrónicos, 2015)

Para qué sirve

Como ha visto hasta ahora, Node está extremadamente bien diseñado para situaciones en que usted esté esperando una gran cantidad de tráfico y donde la lógica del lado del servidor y el procesamiento requeridos, no sean necesariamente grandes antes de responder al cliente. Aquí hay algunos buenos ejemplos en donde Node haría un gran trabajo:

Una API RESTful

Un servicio Web que proporcione una API RESTful toma algunos parámetros, los interpreta, arma una respuesta y descarga esa respuesta (usualmente una cantidad relativamente pequeña de texto) de vuelta al usuario. Esta es una situación ideal para Node, porque puede construirse para que maneje decenas de miles de conexiones. Tampoco requiere una gran cantidad de lógica y básicamente sólo busca valores de una base de datos y los reúne como una respuesta. Como la respuesta es una pequeña cantidad de texto y la solicitud entrante es una pequeña cantidad de texto, el volumen de tráfico no es alto, y una máquina probablemente puede manejar las demandas de API de incluso la API de la más ocupada de las empresas.

Fila de Twitter

Piense en una compañía como Twitter que recibe tweets y los escribe en una base de datos. Literalmente hay miles de tweets llegando cada segundo y la base de datos posiblemente no puede seguir el ritmo del número de escrituras necesarias durante los horarios pico de uso. Node se convierte en una pieza clave de la solución a este problema. Como hemos visto, Node puede manejar decenas de miles de tweets entrantes. Luego puede escribirlos rápida/fácilmente en un mecanismo de cola en memoria (memcached, por ejemplo), desde donde otro proceso separado puede escribirlos en la base de datos.

El rol de Node en esto es reunir rápidamente el tweet y pasar esta información hacia otro proceso responsable de escribirlo. Imagine otro diseño — un servidor PHP normal que intente manejar escrituras en la base de datos misma — cada tweet podría causar una pequeña demora mientras se escribe en la base de datos, dado que el llamado de base de datos estaría bloqueando. Una máquina con este diseño sólo podría manejar 2.000 tweets entrantes por segundo, debido a la latencia de base de datos. A un millón de tweets por segundo, usted estaría hablando de 500 servidores. Node, en cambio, maneja cada conexión y no causa bloqueo, permitiéndole capturar tantos tweets como se le puedan arrojar. Una máquina nodo capaz de manejar 50.000 tweets por segundo, y usted estaría hablando de sólo 20 servidores.

Estadísticas de videojuegos

Si usted alguna vez jugó un juego como Call of Duty on-line, algunas cosas le habrán llamado la atención inmediatamente cuando observó las estadísticas del juego, principalmente el hecho de que deben estar rastreando toneladas de información sobre el juego para poder producir tal nivel de estadísticas. Luego, multiplique esto por los millones de personas que lo juegan en cualquier momento, y tendrá una idea de la inmensa cantidad de información que se genera con bastante rapidez. Node es una buena solución para este escenario, porque puede capturar los datos que están generando los juegos, hacer un mínimo de consolidación con ellos y luego ponerlos en una fila para escribirlos en una base de datos. Parecería algo tonto dedicar todo un servidor a rastrear cuántas balas disparan las personas en los juegos, lo cual podría ser el límite útil si usted utilizara un servidor como Apache, pero parecería menos tonto si en lugar de ello usted pudiera dedicar un solo servidor a rastrear casi todas las estadísticas de un juego, como usted puede llegar a hacerlo con un servidor que ejecute Node. (Abernethy M. 2011)

**2.1.2.1 Node-RED**

Nodo -RED es una herramienta visual para cablear juntos los dispositivos de hardware, APIs y servicios en línea - para el cableado de las cosas del Internet.

Nodo -RED ofrece un editor de flujo basada en navegador que hace que sea fácil de cablear juntos los flujos utilizando los nodos de amplio rango en la paleta. Los flujos pueden ser entonces desplegados en el tiempo de ejecución en un solo clic. Funciones de JavaScript se pueden crear en el editor usando el editor de texto enriquecido. Una biblioteca incorporada le permite guardar funciones útiles, plantillas o fluye para su reutilización.

El tiempo de ejecución de peso ligero se construye sobre Node.js, aprovechando al máximo de sus eventos, modelo sin bloqueo. Esto lo hace ideal para funcionar en el borde de la red en el hardware de bajo costo, como el Raspberry Pi, así como en la nube. (http://nodered.org/docs/)

**2.1.3. Características del dispensador**

Una de las características del prototipo será la funcionalidad de poder alimentar a la mascota de manera inalámbrica, además de un monitoreo del animal para su futura revisión de su alimentación.

Otro sistema parecido al prototipo de esta tesina tiene características como: Plato para comida de perros o gatos de tamaño pequeño o mediano, sistema automático que abre en los intervalos de tiempo una puerta según se programe (6, 12 o 24 horas), el perro o gato nunca se quedará sin alimento, comida para 6 comidas, hecho con material de alta resistencia, fácil de limpiar, cada compartimiento tiene capacidad de 100 gr de comida (Villaba C. 2007).

**2.1.3.1 Servidores para el dispositivo**

**MongoDB** es un sistema de base de datos multiplataforma orientado a documentos, de esquema libre. Como ya os expliqué, esto significa que cada entrada o registro puede tener un esquema de datos diferente, con atributos o “columnas” que no tienen por qué repetirse de un registro a otro. Está escrito en C++, lo que le confiere cierta cercanía al bare metal, o recursos de hardware de la máquina, de modo que es bastante rápido a la hora de ejecutar sus tareas. Además, está licenciado como GNU AGPL 3.0, de modo que se trata de un software de licencia libre. Funciona en sistemas operativos Windows, Linux, OS X y Solaris.

Las características que más destacaría de MongoDB son su velocidad y su rico pero sencillo sistema de consulta de los contenidos de la base de datos. Se podría decir que alcanza un balance perfecto entre rendimiento y funcionalidad, incorporando muchos de los tipos de consulta que utilizaríamos en nuestro sistema relacional preferido, pero sin sacrificar en rendimiento.

MongoDB está escrito en C++, aunque las consultas se hacen pasando objetos JSON como parámetro. Es algo bastante lógico, dado que los propios documentos se almacenan en BSON. Por ejemplo:

*db.Clientes.find({Nombre:"Pedro"});*

La consulta anterior buscará todos los clientes cuyo nombre sea Pedro.

MongoDB viene de serie con una consola desde la que podemos ejecutar los distintos comandos. Esta consola está construida sobre JavaScript, por lo que las consultas se realizan utilizando ese lenguaje. Además de las funciones de MongoDB, podemos utilizar muchas de las funciones propias de JavaSciprt. En la consola también podemos definir variables, funciones o utilizar bucles.

Si queremos usar nuestro lenguaje de programación favorito, existen drivers para un gran número de ellos. Hay drivers oficiales para C#, Java, Node.js, PHP, Python, Ruby, C, C++, Perl o Scala. Aunque estos drivers están soportados por MongoDB, no todos están en el mismo estado de madurez. Por ejemplo el de C es una versión alpha. Si queremos utilizar un lenguaje concreto, es mejor revisar los drivers disponibles para comprobar si son adecuados para un entorno de producción.

**¿Dónde se puede utilizar MongoDB?**

Aunque se suele decir que las bases de datos NoSQL tienen un ámbito de aplicación reducido, MongoDB se puede utilizar en muchos de los proyectos que desarrollamos en la actualidad.

Cualquier aplicación que necesite almacenar datos semi estructurados puede usar MongoDB. Es el caso de las típicas aplicaciones CRUD o de muchos de los desarrollos web actuales.

Eso sí, aunque las colecciones de MongoDB no necesitan definir un esquema, es importante que diseñemos nuestra aplicación para seguir uno. Tendremos que pensar si necesitamos normalizar los datos, denormalizarlos o utilizar una aproximación híbrida. Estas decisiones pueden afectar al rendimiento de nuestra aplicación. En definitiva el esquema lo definen las consultas que vayamos a realizar con más frecuencia.

MongoDB es especialmente útil en entornos que requieran escalabilidad. Con sus opciones de replicación y sharding, que son muy sencillas de configurar, podemos conseguir un sistema que escale horizontalmente sin demasiados problemas. (GenBeta, 2014)

**TCP** (que significa Protocolo de Control de Transmisión) es uno de los principales protocolos de la capa de transporte del modelo TCP/IP. En el nivel de aplicación, posibilita la administración de datos que vienen del nivel más bajo del modelo, o van hacia él, (es decir, el protocolo IP). Cuando se proporcionan los datos al protocolo IP, los agrupa en datagramas IP, fijando el campo del protocolo en 6 (para que sepa con anticipación que el protocolo es TCP). TCP es un protocolo orientado a conexión, es decir, que permite que dos máquinas que están comunicadas controlen el estado de la transmisión.

Las principales características del protocolo TCP son las siguientes:

* TCP permite colocar los datagramas nuevamente en orden cuando vienen del protocolo IP.
* TCP permite que el monitoreo del flujo de los datos y así evita la saturación de la red.
* TCP permite que los datos se formen en segmentos de longitud variada para "entregarlos" al protocolo IP.
* TCP permite multiplexar los datos, es decir, que la información que viene de diferentes fuentes (por ejemplo, aplicaciones) en la misma línea pueda circular simultáneamente.
* Por último, TCP permite comenzar y finalizar la comunicación amablemente.

**El objetivo de TCP**

Con el uso del protocolo TCP, las aplicaciones pueden comunicarse en forma segura (gracias al sistema de acuse de recibo del protocolo TCP) independientemente de las capas inferiores. Esto significa que los routers (que funcionan en la capa de Internet) sólo tienen que enviar los datos en forma de datagramas, sin preocuparse con el monitoreo de datos porque esta función la cumple la capa de transporte (o más específicamente el protocolo TCP).

Durante una comunicación usando el protocolo TCP, las dos máquinas deben establecer una conexión. La máquina emisora (la que solicita la conexión) se llama cliente, y la máquina receptora se llama servidor. Por eso es que decimos que estamos en un entorno Cliente-Servidor.

Las máquinas de dicho entorno se comunican en modo en línea, es decir, que la comunicación se realiza en ambas direcciones.

Para posibilitar la comunicación y que funcionen bien todos los controles que la acompañan, los datos se agrupan; es decir, que se agrega un encabezado a los paquetes de datos que permitirán sincronizar las transmisiones y garantizar su recepción.

Otra función del TCP es la capacidad de controlar la velocidad de los datos usando su capacidad para emitir mensajes de tamaño variable. Estos mensajes se llaman segmentos.

**La función multiplexión**

TCP posibilita la realización de una tarea importante: multiplexar/demultiplexar; es decir transmitir datos desde diversas aplicaciones en la misma línea o, en otras palabras, ordenar la información que llega en paralelo.

Estas operaciones se realizan empleando el concepto de puertos (o conexiones), es decir, un número vinculado a un tipo de aplicación que, cuando se combina con una dirección de IP, permite determinar en forma exclusiva una aplicación que se ejecuta en una máquina determinada. (CCM, 2014)

**2.2 Creación del prototipo**

**2.2.1 Materiales para su elaboración**

\* Raspberry pi. (Se ha decidido utilizar la Raspberry pi debido a gran cantidad de ventajeas que tiene como dice Agustín Díez Devesa en su página unas de sus ventajas son su precio, consumo, espacio y ruido.)

\* Servomotores.

\* Sensor de peso.

\* Contenedor para el alimento.

\* Adaptador Wifi.

\* PermaProtoboard.

\* Cables (Macho/ Macho, Macho/ Hembra).

\*Buzzer.

\*Leds.

\*Switch (pulsador o interruptor).

**2.2.2 Pasos a seguir para su construcción**

Montar los servos:

Los servos de rotación continua necesitan ser unido a las asas para hacer girar la rueda de la aleta. Primero lo primero, prueba para ver si el servo es lo suficientemente potente como para mover la palanca. Cogí un poco de cinta adhesiva en este caso, y grabé a la parte delantera del mango, y aferré a él, mientras le contaba el servo para funcionar.



* Montar ServosIMG\_3434.jpg. http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

Luego los tiradores deben estar en la parte de atrás, para que no estén expuestos a alguien golpear o chocar ellos (o quedar atrapados). Así que me mudé las asas a la parte de atrás. Al mover las asas en la parte posterior, que necesitaba para eliminar aproximadamente 3/4 "en cada extremo. De esta manera el mango puede girar fácilmente en un círculo 360deg.



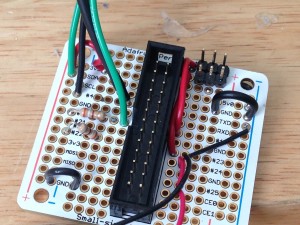
2,3. Moviendo asas.IMG\_3444.jpg,IMG\_3429.jpghttp://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

Después cortamos un canal para la cabeza del servo. Finalmente me acabé cortando unos 2 mm en el canal, y, básicamente, hacer un esquema para que el servo debe sentarse. Esto era sobre todo por lo que cuando me pego a este tipo con pegamento caliente, que tiene algo que empujar contra, en tanto que si se coloca en la superficie, no tendría el apalancamiento y que rompería con gran rapidez. El pegamento caliente es ideal para la celebración de las cosas en su lugar, pero no para hacer fuertes uniones mecánicas. También he añadido dos bandas de sujeción y de apoyo después de que el pegamento caliente había empezado a ponerse.



4. Pegar los Servos.IMG\_3451.jpg.http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

Utilizando un " Pequeño Perma - proto Raspberry Pi Breadboard PCB Kit " de Adafruit en este caso. Comience agregando un encabezado de tres clavijas a la placa para cada servo. Esto hace que sea fácil de conectar / desconectar los servo



5. Cableado en PermaProto.IMG\_3494.jpg.http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

Ahora, una vez que todo está soldado, y ha probado todo, ahora usted puede comenzar a modificar su contendor del alimentador, se adhiere todo de nuevo. Me perforó agujeros para el cable de alimentación USB, los encabezados de servo, un a través del agujero, y los botones laterales.



6. Acomodo del cableado.IMG\_3509.jpg.http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

Por último Guardar todo en el pilar del dispensador.



7. Cableado-Pilar.IMG\_3510.jpg.http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

**2.2.3 Programación adecuada:**

Se utilizara en este caso los lenguajes de programación los cuales según Sara Álvarez (2006) "Es un conjunto de reglas semánticas así como sintácticas que los programadores usan para la codificación de instrucciones de un programa o algoritmo de programación".

Hay muchos lenguajes de programación por nombrar los más relevantes o los que hemos trabajado los cuales son:

**- C#.**

C# es un lenguaje de programación que se ha diseñado para compilar diversas aplicaciones que se ejecutan en .NET Framework. C# es simple, eficaz, con seguridad de tipos y orientado a objetos. Las numerosas innovaciones de C# permiten desarrollar aplicaciones rápidamente y mantener la expresividad y elegancia de los lenguajes de estilo de C.

Visual C# es una implementación del lenguaje C# de Microsoft. Visual Studio ofrece compatibilidad con Visual C# con un completo editor de código, un compilador, plantillas de proyecto, diseñadores, asistentes para código, un depurador eficaz y de fácil uso y otras herramientas. La biblioteca de clases de .NET Framework ofrece acceso a numerosos servicios de sistema operativo y a otras clases útiles y adecuadamente diseñadas que aceleran el ciclo de desarrollo de manera significativa. (Microsoft, 2015)

- **Java**. Este es uno de los más famosos según Martin Ángel A. (2001) "Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado.” (Miguel A., 2001)

-**Python**.

Básicamente, Python es un lenguaje de programación de alto nivel, interpretado y multipropósito. En los últimos años su utilización ha ido constantemente creciendo y en la actualidad es uno de los lenguajes de programación más empleados para el desarrollo de software.

Python puede ser utilizado en diversas plataformas y sistemas operativos, entre los que podemos destacar los más populares, cómo Windows, Mac OS X y Linux. Pero, además, Python también puede funcionar en Smartphone, Nokia desarrolló un intérprete de esté lenguaje para su sistema operativo Symbian.

¿Tiene Python un ámbito específico? Algunos lenguajes de programación sí que lo tienen. Por ejemplo, PHP fue ideado para desarrollar aplicaciones Web. Sin embargo, esté no es el caso de Python. Con este lenguaje podemos desarrollar software para aplicaciones científicas, para comunicaciones de red, para aplicaciones de escritorio con interfaz gráfica de usuario (GUI), para crear juegos, para Smartphone y por supuesto, para aplicaciones web.

Empresas y organizaciones del calibre de Industrial Light & Magic, Walt Disney, la NASA, Google, Yahoo!, Red Hat y Nokia hacen uso intensivo de esté lenguaje para desarrollar sus productos y servicios. Esto demuestra que Python puede ser utilizado en diversos tipos de sectores, con independencia de su actividad empresarial.

Entre las principales razones para elegir Python, son muchos los que argumentan que sus principales características lo convierten en un lenguaje muy productivo. Se trata de un lenguaje potente, flexible y con una sintaxis clara y concisa. Además, no requiere dedicar tiempo a su compilación debido a que es interpretado.

Python es open source, cualquiera puede contribuir a su desarrollo y divulgación. Además, no es necesario pagar ninguna licencia para distribuir software desarrollado con esté lenguaje. Hasta su intérprete se distribuye de forma gratuita para diferentes plataformas.

La ùltima versión de Python recibe varios nombres, entre ellos, Python 3000 y Py3K, aunque, habitualmente, se le denomina simplemente Python 3. (Santana C., 2013)

**- JavaScript**

¿Qué es JavaScript?

JavaScript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con JavaScript podemos crear diferentes efectos e interactuar con nuestros usuarios.

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

¿Es compatible con navegadores?

JavaScript es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera, Mozilla Firefox, entre otros.

Con el surgimiento de lenguajes como PHP del lado del servidor y JavaScript del lado del cliente, surgió Ajax en acrónimo de (Asynchronous JavaScript And XML). El mismo es una técnica para crear aplicaciones web interactivas. Este lenguaje combina varias tecnologías:

HTML y Hojas de Estilos CSS para generar estilos.

Implementaciones ECMAScript, uno de ellos es el lenguaje JavaScript.

XMLHttpRequest es una de las funciones más importantes que incluye, que permite intercambiar datos asincrónicamente con el servidor web, puede ser mediante PHP, ASP, entre otros. (Pérez D., 2007)

**2.3 Funcionamiento del prototipo**

El sistema funciona por programación de tareas (SystemScheduler), este mecanismo funciona de la siguiente manera: SystemScheduler es más fácil de usar que el programador de Windows y también le permite entrar parámetros para un programa o archivo. Específicamente, le permite entrar “presiones de teclado” en una manera fácil de entender, muy útil para no programadores. Por ejemplo, cuando se quiere seleccionar un área para desfragmentar un disco, o un área de disco específica para revisar con el antivirus, las cadenas de teclas se pueden programar para que el programa opere independiente de la presencia del usuario (Camillo J.2015).

El usuario se conecta por medio de una aplicación web o móvil y ejecuta el sistema para que por medio de la red mande una señal al dispositivo Raspberry y este a su vez, dispensara la comida.

**2.3.1 Programación de tareas (Cron)**

**¿Qué es cron?**

Cron es el nombre del programa que permite a usuarios Linux/Unix ejecutar automáticamente comandos o scripts (grupos de comandos) a una hora o fecha específica. Es usado normalmente para comandos de tareas administrativas, como respaldos, pero puede ser usado para ejecutar cualquier cosa. Como se define en las páginas del manual de cron (**#>man cron**) es un demonio que ejecuta programas agendados.

En prácticamente todas las distribuciones de Linux se usa la versión Vixie Cron, por la persona que la desarrolló, que es Paul Vixie, uno de los grandes gurús de Unix, también creador, entre otros sistemas, de BIND que es uno de los servidores DNS más populares del mundo.

El comando crontab se utiliza en sistemas UNIX para programar la ejecución de otros comandos, es decir, para automatizar tareas. Podemos ver los crontabs que se están programados y también editarlos, lógicamente.

Para verlos, utilizamos este comando:

Sudocrontab -l

Sudocrontab -l

Para editarlos:

Sudocrontab -e

Sudocrontab -e

FORMATO DE LAS TAREAS

Las tareas cron siguen una determinada sintaxis. Tienen 5 asteriscos seguidos del comando a ejecutar. Ahora explicaré para qué sirve cada cosa.

\* \* \* \* \* /bin/ejecutar/script.sh

\* \* \* \* \* /bin/ejecutar/script.sh

**Usando cron**

Hay al menos dos maneras distintas de usar cron:

La primera es en el directorio etc., donde muy seguramente encontrarás los siguientes directorios:

* cron.hourly
* cron.daily
* cron.weekly
* cron.monthly

Si se coloca un archivo tipo script en cualquiera de estos directorios, entonces el script se ejecutará cada hora, cada día, cada semana o cada mes, dependiendo del directorio.

Tabla 1. Tabla de control

|  |  |
| --- | --- |
| **Campo** | **Descripción** |
| Minuto | Controla el minuto de la hora en que el comando será ejecutado, este valor debe de estar entre 0 y 59. |
| Hora | Controla la hora en que el comando será ejecutado, se especifica en un formato de 24 horas, los valores deben estar entre 0 y 23, 0 es medianoche. |
| Día del Mes | Día del mes en que se quiere ejecutar el comando. Por ejemplo se indicaría 20, para ejecutar el comando el día 20 del mes. |
| Mes | Mes en que el comando se ejecutará, puede ser indicado numéricamente (1-12), o por el nombre del mes en inglés, solo las tres primeras letras. |
| Día de la semana | Día en la semana en que se ejecutará el comando, puede ser numérico (0-7) o por el nombre del día en inglés, solo las tres primeras letras. (0 y 7 = domingo) |
| Usuario | Usuario que ejecuta el comando. |
| Comando | Comando, script o programa que se desea ejecutar. Este campo puede contener múltiples palabras y espacios. |

**2.4 Diferentes ejemplos de la idea:**

Uno de las variaciones del dispensador es el CleverPet, Su función es dar pequeñas cantidades de alimento al perro por seguir un patrón de botones los cuales él debe de presionar. En un articula de la página del producto dice: " Con el software inteligente integrado por doctores en neurociencia computacional, CleverPet está diseñado para satisfacer las necesidades cognitivas de su perro. Está optimizado para el compromiso del perro. Haciendo referencia a que cada vez que el perro completa una sesión esta cambia para que el perro sigua jugando contando como ejercicio.

Otro es el Petnet un dispensador más sofisticado para hacer que la mascota tenga dietas más equilibradas. AnuSaptharishi (2014) dice en su blog que: "No todos los comederos para mascotas son creados iguales. Con tantos alimentadores que dicen ser una solución para la alimentación de su mascota mientras usted está fuera de casa, puede ser confuso para encontrar la mejor solución para su amada mascota. La única certeza es que los alimentadores de automáticos regulares no proporcionan la funcionalidad de alimentación avanzada del Petnet( io) SmartFeeder .

**Dispensadores para animales callejeros**

Pugedon es un proyecto turco que busca ayudar a los perros callejeros reciclando botellas, ya que por cada botella de agua que alguien recicla, suministra bebida y comida a los animales callejeros, su sustento diario. Esta máquina tiene todo su sentido en Turquía ya que son más de 100.000 perros y gatos callejeros los que suelen deambular por sus calles.

La máquina que han instalado, incorpora un contenedor para reciclado de botellas. Por cada botella de plástico que se recicla, por un dispensador sale una ración de comida y bebida para los animales. El agua que sobra se vierte por un hueco que tiene la máquina; esta cae en un recipiente en el que los animales de la calle pueden beber.

Así como en grandes concentraciones urbanas como Pekín y la ciudad de México se implementaron medidas sustentables en el transporte público que, además, representaran beneficios en la economía de los usuarios: una nueva forma de pago con botellas pet en máquinas recicladoras, en Turquía, una empresa llamada Pugedon desarrolló una máquina que, del mismo modo que en China y México, funciona con botellas de plástico, pero en lugar de hacer posible un viaje del usuario, los beneficios son para los perros callejeros.

Con cada botella de plástico depositada, la máquina libera agua y alimento para los perros que circulan por las calles de Estambul. ThePugedon Smart Recycling Boxes operan sin costo alguno, y los ingresos obtenidos de reciclar las botellas son destinados para cubrir el costo de la comida.

Esta iniciativa ve por el ambiente y considera un beneficio para los muchos perros de la calle que dependen de la caridad de los habitantes.

Es bien sabido que Turquía cuenta con una importante población de perros callejeros en las áreas urbanas. Sólo la ciudad de Estambul es hogar de más de 150 mil perros y gatos, de acuerdo con Deutsche Welle. Esta situación provoca, por un lado, el interés por resolver las necesidades básicas a los caninos, y por otro, la animadversión de quienes consideran a estos como un peligro y fuente de enfermedades.

**2.5 Beneficios de su implementación**

Al implementar el dispensador de comida para mascotas, la alimentación de la mascota se verá mejor organizada y así mismo se podrá dar alimento estando fuera del hogar utilizando una aplicación web o móvil. Este dispositivo es muy útil si es recurrente alguna salida fuera de la cuidad y dejar a la mascota sola en casa.

Ventajas

* Mejor alimentación para la mascota
* Inalámbrico
* Aplicación web y móvil
* Alimentar a la mascota fuera del hogar
* Futuras adaptaciones o mejoras

**2.6 Costos y tiempo**

El total de todo el material da un precio aproximado de $2,500. Los precios individuales de cada material son los siguientes:

Tabla 2. Tabla de costos

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Cantidad** | **Precio** | **Total** |
| Dispensador de alimentos | 1 | 39.99 | 39.99 |
| Raspberry Pi | 1 | 35 | 35 |
| Adaptador Wifi | 1 | 5 | 5 |
| Kit de PCB para tablero de pruebas Perma-Proto Raspberry Pi de tamaño pequeño de Adafruit | 1 | 4.95 | 4.95 |
| Servomotor de rotación continua | 2 | 14 | 28 |
| Pines de cabecera para Servomotores | 1 | 3 | 3 |
| Cable conector | 1 | 8.49 | 8.49 |
| Resistores de 380 Ohm | 4 | 0.05 | 0.2 |
| Fuente de alimentación USB de 10 Watt 5V | 1 | 5 | 5 |
| Buzzer | 1 | 3.99 | 3.99 |
| Interruptores con LED para alimentación manual | 2 | 4.99 | 9.98 |
| LEDS de preferencia propia | 2 | 0.25 | 0.5 |
| Tubo de PVC para boquillas de alimentación | 2 | 1.65 | 3.3 |
| Cinchos | 8 | 0.05 | 0.4 |
| Cinta adhesiva | 1 | 0 |  |
| tubo termoretractil | 1 | 0 | 0 |
| **Total** |  |  | **147.8** |
|  |  |  |  |

tabla de materiales.spreadsheets.jpg.http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

El tiempo empleado construir el dispositivo teniendo todos los materiales es de 1 semana.

**2.7 Adaptaciones futuras**

Cámara Pi: El módulo de la cámara Raspberry Pi se puede utilizar para tomar video de alta definición, así como alambiques fotografías. Es fácil de utilizar para los principiantes, pero tiene mucho que ofrecer a los usuarios avanzados si usted está buscando para ampliar su conocimiento. Hay un montón de ejemplos en línea de personas que lo usan para time-lapse, cámara lenta y otra inteligencia vídeo. También puede utilizar las bibliotecas que Bundle con la cámara para crear efectos.

Si usted está interesado en el meollo de la cuestión, usted querrá saber que el módulo tiene una cámara de foco fijo de cinco megapíxeles que soporta 1080p30, 720p60 y vídeo VGA90 modos, así como imágenes fijas de captura. Se une a través de un cable plano de 15 cm hasta el puerto de CSI en la Raspberry Pi.

La cámara funciona con todos los modelos de Raspberry Pi 1 y 2. Se puede acceder a través de las API MMAL y V4L, y hay numerosas bibliotecas de terceros creadas para ello, incluyendo el Picamera biblioteca de Python.

El módulo de la cámara es muy popular en aplicaciones de seguridad en el hogar y en la vida silvestre cámaras trampa.

* Bocinas
* Clips de audio
* Módulo de dispensador de agua: tendrá una función parecida pero dispensara agua a un tazón diferente para que la mascota pueda tomar agua.

Tipos

Algunos dispensadores de agua son unidades autónomas con grandes reservas de agua. Otros están conectados al suministro de agua. Estos por lo general tienen un filtro instalado para filtrar el agua del grifo.

Opciones

Los dispensadores de agua pueden proporcionar grifos de agua caliente y fría. Los modelos más simples facilitan sólo uno o el otro. Algunos tienen un tercer grifo de agua a temperatura ambiente.

Usos

Los dispensadores de agua vienen en diferentes capacidades. Algunos son útiles para algunas personas en el hogar, mientras que otros sirven a un gran número de personas en una oficina. Un dispensador conectado a un suministro de agua funciona en cualquier capacidad.

Beneficios

Los dispensadores de agua producen menos residuos que comprar botellas de agua individual. También pueden aliviar las preocupaciones acerca de los productos químicos presentes en las botellas de plástico o impurezas presentes en el agua del grifo.

Consideraciones

Tendrás que comprar filtros para el dispensador que está conectado a la línea de agua o comprar los grandes contenedores de agua filtrada para las unidades independientes. Cualquier opción cuesta menos en el tiempo que la compra de botellas de agua individuales.

**CAPITULO III**

**MARCO METODOLOGICO**

**3.1 DEFINICION DEL TIPO DE INVESTIGACION A REALIZAR**

El tipo de investigación es experimental. El autor Marín (2008) dice que la investigación experimental es la que obtiene su información de la actividad intencional realizada por el investigador y que se encuentra dirigida a modificar la realidad con el propósito de crear el fenómeno mismo que se indaga, y así poder observarlo, sobre la misma se puede o no usar un grupo de control, con el fin de hacer las comparaciones necesarias para comprobar las hipótesis o rechazarlas según el caso. Por lo tanto nuestro proyecto se desarrolla en un ambiente experimental ya que involucra varios prototipos y diferentes variables de su posible aplicación, si los resultados arrojados favorecen a su implementación o no.

2.- En su categoría: Experimento puro. El autor Avalos (2010) dice que consiste en la manipulación de variables independientes para ver sus efectos sobre las variables dependientes en una situación de control. Por lo cual, las causas y efectos del proyecto se tendrán que evaluar.

**3.2 Población y Muestra**

**Población**

La población de esta investigación son los hogares de la población de Los Mochis, Sinaloa.

**Muestra**

La determinación de la muestra fue por: Probabilidad. En su tipo no probabilística. Por lo cual se realizara encuesta a 20 Personas de la población de los Mochis, Ahome Sinaloa. Debido a que los sujetos a entrevistar han presentado los mismos problemas de alimentar a sus mascotas cuando ellos están ausentes.

**3.3 METODOS DE INVESTIGACION**

El método de investigación es empírico. El autor Domínguez (2013) dice que la ***empírico***. (del lat. Empiricus, y este del gr. ἐµpeιρικόσ) que se rige por la experiencia. Los datos empíricos son sacados de las pruebas acertadas y de los errores cometidos en el proceso. Por lo cual, los datos obtenidos durante el conocimiento de prototipo se usaran para las pruebas en su funcionamiento.

En su categoría: Método Experimental. El autor Hernández (2010) dice que algunos lo consideran por su gran desarrollo y relevancia un método independiente del método empírico, considerándose a su vez independiente de la lógica empírica su base, la lógica experimental. Por lo cual, al desarrollar y profundizar en el enfoque del prototipo se hace un mejor conocimiento empírico sobre el mismo.

Se usaran los siguientes procedimientos: Análisis y abstracción ya que el proceso de creación del prototipo se basa en conocer las partes y cualidades que lo componen, así como las diferentes relaciones y su propiedad de las cosas y fenómenos.

**3.4 OPERACIONALIZACION DE LAS VARIABLES**

Para elaborar el diseño del dispensador de alimentos de comida para mascota que funcione con remotamente con conexión de internet usando una Raspberry Pi se ocuparan los siguientes materiales: dispensador de comida, Microcomputador Raspberry Pi, adaptador Wi-Fi, una protoboard, servomotor de rotación continua, conectores macho para los servos, resistencias, cable conector, fuente de alimentación, buzzer, leds, tubo de pvc, cinchos. Teniendo ya los materiales empezamos instalando en la Raspberry pi los programas de la compatibilidad del software (raspbian), el adaptador de wifi y el programa en sí de cómo va a funcionar. Después comenzaremos a montar el servo-motor en el dispensador de comida usando y pegándolos al dosificador y le agregamos unos cinchos para que quede fijo y no se mueva o caiga al momento de girar. El siguiente paso es hacer el cableado utilizando la protoboard pequeña de la Raspberry pi a la mini computadora, a los servo motores y a los LED´s, y el buzzer que estarán dentro del dispensador. En el hueco por donde dispensa la comida se colocara el tubo de PVC para que la comida caiga en el plato de comida de la mascota. Su elaboración tomara de un día a cinco días hábiles dependiendo de la disposición de tiempo de la persona y su costo total por todos los materiales serán de un estimado de $ 2,500.00 pesos.

Tabla 3. Tabla de variables y operaciones

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Definición Conceptual | Definición Operacional | Indicadores |
| Tiempo de elaboración | Horas que los integrantes le dedicaran en la realización del dispensador. | Cantidad de horas dedicadas en la elaboración con la presencia de todos los integrantes. | - Recolección de los materiales.  - Ensamblaje de los materiales.  - Realización del cableado. |
| Dispensador de comida que funcione remotamente | El dispensador funcionara vía remota wifi en respuesta a una aplicación que se encontrara instalada en el celular. | Realización y prueba de la adaptación wifi del dispensador y realización de la aplicación de celular y probar su compatibilidad. | -Realización de la adaptación wifi.  -Elaboración de la aplicación.  -Sincronizar el comedero y la aplicación. |
| Costo de elaboración | Recolección del dinero necesario de entre todos los integrantes. | Total monetario que todos los integrantes necesitaran para cubrir los costos de los materiales. | -Recolección del dinero necesario. |

**3.5 Instrumentos de medición**

El instrumento de medición utilizado es: Encuesta. El autor: Richard L. Sandhusen (2015) las encuestas obtienen información sistemáticamente de los encuestados a través de preguntas, ya sea personales, telefónicas o por correo. Por lo cual es apto por: porque es el medio más factible de obtener respuestas acerca del proyecto.

**3.6 Prueba piloto**

La aplicación de la prueba piloto se realizara a \_\_\_\_\_\_\_\_\_\_\_\_, se llevó a cabo previo a la aplicación general del instrumento de medición en la población muestra. Los resultados obtenidos son:

**3.7 procesamiento de datos**

El procesamiento de datos de esta investigación se llevara con: Conclusiones. Se dará la conclusión de todas las respuestas dadas en base a la encuesta realizada y envase a los datos obtenidos.

**CAPITULO IV**

**ANÁLISIS DE RESULTADOS Y PROPUESTA**

**4.1 Análisis de los resultados**

Uno de los principales objetivos a cumplir con esta investigación era conocer los beneficios que proporcionaría la implementación de dicho sistema en los hogares, por lo cual se optó con encuestar a la población que tiene mascotas y los resultados fueron que el 90% de las personas encuestas consideran que este dispositivo si sería de gran ayuda y les resolvería los problemas de alimentar a su mascota, ya que el 60% de ellos, reconoció haber olvidado en alguna ocasión alimentar a sus mascotas por diversos motivos.

Conocer los materiales, costos y tiempo de implementación del prototipo.

Los materiales que se utilizaran para la realización del dispensador de comida para mascotas son:

\* Raspberry pi.

\* Servomotores.

\* Sensor de peso.

\* Contenedor para el alimento.

\* Adaptador Wifi.

\* PermaProtoboard.

\* Cables (Macho/ Macho, Macho/ Hembra).

\*Buzzer.

\*Leds.

\*Switch (pulsador o interruptor).

Los siguientes materiales están basados en esta lista:

Tabla 4. Tabla de costos

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Cantidad** | **Precio** | **Total** |
| Dispensador de alimentos | 1 | 39.99 | 39.99 |
| Raspberry Pi | 1 | 35 | 35 |
| Adaptador Wifi | 1 | 5 | 5 |
| Kit de PCB para tablero de pruebas Perma-Proto Raspberry Pi de tamaño pequeño de Adafruit | 1 | 4.95 | 4.95 |
| Servomotor de rotación continua | 2 | 14 | 28 |
| Pines de cabecera para Servomotores | 1 | 3 | 3 |
| Cable conector | 1 | 8.49 | 8.49 |
| Resistores de 380 Ohm | 4 | 0.05 | 0.2 |
| Fuente de alimentación USB de 10 Watt 5V | 1 | 5 | 5 |
| Buzzer | 1 | 3.99 | 3.99 |
| Interruptores con LED para alimentación manual | 2 | 4.99 | 9.98 |
| LEDS de preferencia propia | 2 | 0.25 | 0.5 |
| Tubo de PVC para boquillas de alimentación | 2 | 1.65 | 3.3 |
| Cinchos | 8 | 0.05 | 0.4 |
| Cinta adhesiva | 1 | 0 |  |
| tubo termoretractil | 1 | 0 | 0 |
| **Total** |  |  | **147.8** |

8.-tabla de materiales.spreadsheets.jpg.http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/

Comparar el sistema con otros proyectos ya desarrollados.

La principal función del modelo a realizar es que el dueño(a) pueda alimentar a su mascota vía remota enviando una señal de cuantos gramos le desea dispensar. Otros de los dispensadores vistos como el cleverpet el cual su función es premiar a la mascota tras seguir una serie de patrones de botones los cuales se iluminan y la mascota debe de presionarlos siguiente el patrón si lo cumple obtiene el premio, cada vez que cumple un patrón aparee otro más complejo que el anterior lo cual permite al dueño mantener activo a su perro. En otro caso con el petnet es un dispensador remoto que tiene un funcionamiento muy similar al nuestro pero su principal énfasis es en sí es que el dueño pueda proporcionar una dieta más balanceada a su mascota controlando los gramos al día que le da a la mascota.

Descripción de funcionamiento.

Una vez ya ensamblado el prototipo funcionara de la siguiente forma: El dueño tendrá acceso a la aplicación hecha para el dispensador y pondrá cuantos gramos de alimento desea alimentar a su mascota, después el dueño oprimirá un botón el cual enviara una señal al wifi que esté conectado a la placa Raspberry pi. Tras recibir la señal la Raspberry pi accionara el motor el cual hará girar la mescladora permitiendo que la comida caiga en el plato. El plato tendrá un sensor de peso el cual se encargara de calcular el peso en gramos de alimento realizado por el dueño. Ya realizado la placa regresara un mensaje al correo del usuario para notificarle que su mascota fue alimentada.

La mayoría de los dispensadores tienen un funcionamiento parecido ya que el usuario pone en la aplicación la cantidad de comida y los envía vía remota al dispensador solo que con las variaciones ya mencionadas anteriormente.

Verificar si es factible

Quisimos ver si era factible implementar el dispensador en los hogares, con los resultados obtenidos de las encuestas el 90% de los encuestados dicen que les resolvería los problemas de alimentar a su mascota en su hogar, en otra pregunta les preguntamos si les parecía interesante que de que el dispositivo funcionara con internet de vuelta recibimos un buen resultado de que 90% dio positivo al dispensador. Con todo esto el impacto del

**4.2 PROPUESTA**

La propuesta acerca de esta tesis se basa en los beneficios que proporcionara el dispensador de alimento para mascotas vía remota, es que beneficie tanto a la persona o el dueño y a la mascota misma, ya que en base a respuesta de los encuestados por los pendientes o trabajos que se tiene se puede olvidar alimentar a su mascota.

En otro caso si se encuentra fuera de la ciudad o de la casa poder descartar la opción de pedirle a un vecino o algún otro familiar que lo alimente por ti, pero como saber si la persona lo alimenta o le está dando la cantidad de alimento que el animal coma, ahí entrara el dispensador de alimentos vía remota.

Su conexión vía wifi permitirá que reciba la señal de la aplicación de teléfono móvil en el lugar al que vaya la persona.

Desde la aplicación el dueño podrá especificar cuantos gramos de alimento la dará a su mascota, cuantas veces al día comerá y la hora que usualmente se alimenta a la mascota.

**CONCLUSIONES**

* A la pregunta general: ¿Cómo diseñar e implementar un sistema dispensador de comida para mascotas que funcione inalámbricamente por medio de la conexión a internet utilizando los recursos de una microcomputadora Raspberry Pi? se dio a conocer los resultados mediante encuestas, por lo cual los resultados obtenidos muestran que la hipótesis es verdadera.

De los objetivos específicos se cumplieron:

* Conocer los materiales, costos y tiempo de implementación del prototipo. Este objetivo se cumplió debido a la investigación realizada de los materiales que se necesitaran para su elaboración, los costos que exigen estos materiales y el tiempo que toma elaborar dicho dispositivo.
* Comparar el sistema con otros proyectos ya desarrollados. Durante la investigación se puso a comparación diferentes dispositivos con el prototipo de esta tesina. Se comparó desde su sistema hasta el hardware.
* Describir el funcionamiento de dispensadores controlados remotamente. En esta tesina se describe el funcionamiento del prototipo, así como de otros en general.
* Verificar si es factible implementar dispensadores remotos en los hogares. Debido a los resultados que obtuvimos en las encuestas se dio a conocer a través de las personas que el 90% vio factible implementar dicho dispositivo en su hogar.

De los objetivos específicos no se cumplieron:

* Conocer los beneficios que proporcionaría la implementación de dicho sistema en los hogares.

Debido a que la investigación realizada está en proceso de elaboración del dispensador, no se conocerá aun los beneficios de la implementación de dicho dispensador.

* Conocer el impacto de la implementación en los hogares de la cuidad de Los Mochis.

Como se realiza un prototipo se deberá de probar y adaptar para que su uso sea apropiado para las personas, debido a que es un prototipo en proceso de elaboración no se ha puesto a prueba en un hogar.

Conforme a los resultados obtenidos por medio de las encuestas, resultados de otros dispensadores, otras investigaciones de campo y en base a otras hipótesis, nuestra tesis es aceptada.

**RECOMENDACIONES**

Con los resultados obtenidos durante la investigación a futuro se puede implementar un dispositivo que tenga las mismas funcionalidades pero con más características. Este dispositivo funciona para mascotas terrestres como perros o gatos pero a futuro se puede pensar en darle un giro a la idea cambiando las mascotas terrestres por las aéreas o acuáticas como los peces, tortugas, pájaros.

La tecnología usada en este dispositivo funciona para muchas más cosas, el dispositivo Raspberry Pi funciona como una minicomputadora basado en diferentes sistemas operativos. Se podría adecuar una aplicación móvil para hacer funcionar el dispositivo de manera remota.

Se podría adecuar un dispensador de agua junto al dispositivo para hacerlo más novedoso y que cumpliera con las diferentes necesidades de la mascota. Así mismo con un sensor de presión para calcular cuanta comida hay dentro del contenedor, de igual manera con el dispensador de agua.

**REFERENCIAS**

Abernethy M. (2011, Julio). ¿Simplemente qué es Node.js? Recuperado el 23 de octubre de 2015 de https://www.ibm.com/developerworks/ssa/opensource/library/os-nodejs/)

Agustín D. (2014, Agosto). Ventajas de una Raspberry pi. Recuperado el 16/10/2015 de <http://adiezdevesa.com/ventajas-de-una-raspberry-pi/>

Alegasa (2014, noviembre) Diccionario de informática y tecnología, Definición de inalámbrico. Recuperado el 15 de octubre del 2015 de <http://www.alegsa.com.ar/Dic/inalambrico.php#sthash.VcVE9ZQq.dpuf>

Avalos (201, Octubre). Experimento puro. Recuperado el 29/10/2015, de http://es.slideshare.net/metodospsicologia/sintesis-experimento-puro

Benítez M. (2015, Enero). Raspberry Pi, una consola de juegos con RetroPie. Recuperado el 23 de octubre de 2015 de <http://www.babuleando.com/raspberry-pi-una-consola-de-juegos-con-retropie/>

Bryan. D (2013, diciembre).Raspberry Pi Power Cat Feeder – Updates. Recuperado el 10 de septiembre del 2015 de<http://drstrangelove.net/2013/12/raspberry-pi-power-cat-feeder-updates/>

Camillo J (2015, Septiembre) Programador de tareas, System Schedule. Recuperado el día 16 de octubre de 2015 de <http://www.deltaasesores.com/recursos/software/3459-programador-de-tareas>

CCM (2014, Septiembre) Protocolo TCP. Recuperado el 24 de octubre de 2015 de http://es.ccm.net/contents/281-protocolo-tcp

Domiguez (2013, Marzo). Investigación empírica. Recuperado el 29/10/2015, de

<http://www.uv.mx/personal/clelanda/files/2013/03/02-Metodos-investigacion-empirica-y-teorica.pdf>

EcoInventos (2014, Agosto) La máquina que alimenta perros callejeros reciclando botellas. Recuperado el 23 de octubre de 2015 de http://ecoinventos.com/la-maquina-que-alimenta-perros-callejeros-reciclando-botellas/#ixzz3pRKLjSMa

Foco Blanco (2014, Agosto) Dispensadores de comida para perros y gatos callejeros a cambio de botellas de plástico. Recuperado el 23 de octubre de 2015 de http://focoblanco.com.uy/2014/07/dispensadores-de-comida-para-perros-y-gatos-callejeros-cambio-de-botellas-de-plastico/

GenBeta (2014, Febrero) MongoDB: qué es, cómo funciona y cuándo podemos usarlo (o no). Recuperado el 24 de octubre de 2015 de http://www.genbetadev.com/bases-de-datos/mongodb-que-es-como-funciona-y-cuando-podemos-usarlo-o-no

Getcleverpet. (Recuperado en 2015, Octubre). MeetCleverPet. Recuperado el 16/10/2015, de <http://getcleverpet.com/>

Gonzales S. (2005). Manual básico de cron. Recuperado el día 23 de octubre de 2015 de <http://www.linuxtotal.com.mx/?cont=info_admon_006>

Hernandez. (2010, Octubre). Método experimental. Recuperado el 29/10/2015, de

http://tesistesina.blogspot.mx/2010/08/metodo-cientifico\_12.html

Marín L. (2008, Marzo). Clasificación de la investigación. Recuperado el 29/10/2015, de <https://metinvestigacion.wordpress.com/>

Madison A. (2015, Octubre) ¿Cuáles son las funciones de un dispensador de agua? Recuperado el 23 de octubre de 2015 de http://www.ehowenespanol.com/cuales-son-funciones-dispensador-agua-hechos\_400583/

Microsoft (2015, Octubre) Visual C#. Recuperado el día 23 de octubre de 2015 de https://msdn.microsoft.com/es-es/library/kx37x362.aspx?f=255&MSPPError=-2147217396

Miguel A. (2001, Julio). Qué es Java, Descripción y características de este potente y moderno lenguaje de programación., Recuperado el 16/10/2015, de <http://www.desarrolloweb.com/articulos/497.php>

Node-RED (2015, Octubre). Node-RED. Recuperado el día 23 de octubre de 2015 de http://nodered.org/docs

Owano, N. (2012, Julio) Raspberry Pi gets customized OS called Raspbian. Recuperado el 15 de octubre del 2015 de <https://es.wikipedia.org/wiki/Raspberry_Pi#Software>

Pastor J. (2015, Agosto). Qué puedes y qué no puedes hacer con Windows 10 IoT Core en las Raspberry Pi 2. Recuperado el 23 de octubre de 2015 de http://www.xataka.com/makers/que-puedes-y-que-no-puedes-hacer-con-windows-10-iot-core-en-las-raspberry-pi-2

Pérez D. (2007, Julio). ¿Qué es JavaScript? Recuperado el día 23 de octubre de 2015 de http://www.maestrosdelweb.com/que-es-javascript/

Pet Net (2014, Agosto).Smart Feeder vs. Regular Pet Auto Feeders.Recuperado el día 15 de octubre del 2015 de http://www.petnet.io/pet\_health\_blogs/smartfeeder-vs-regular-pet-auto-feeders#.ViEZJX4vfIU

RaspberryPi (2013, Diciembre) Raspberry Pi Powered Cat \_ Human Feeder. Recuperado el día 16 de octubre del 2015 de <https://docs.google.com/spreadsheets/d/1c5p9tFtqQc7_e5I8v1sYqzyVALdSKkCW0qujq4JIrwQ/edit#gid=0>

Raspberry Pi Foundation (2015, Octubre) Módulo Cámara. Recuperado el 24 de octubre de 2015 de https://www.raspberrypi.org/products/camera-module/

Raspberry Pi Foundation (2015, octubre) Raspberry Pi, What is a Raspberry Pi?.Recuperado el 15 de octubre del 2015 de<https://www.raspberrypi.org/help/what-is-a-raspberry-pi/>

Raspbian (2015, Octubre). Raspbian. Recuperado el 23 de octubre de 2015 de https://www.raspbian.org/

Santana C. (2013, Marzo) ¿Qué es Python? Recuperado del día 23 de octubre de 2015 de https://www.codejobs.biz/es/blog/2013/03/02/que-es-python

Sara A. (2006, Febrero). Conceptos básicos y definiciones sobre programación, Recuperado el 16 de octubre del 2015 de <http://www.desarrolloweb.com/articulos/2357.php>

Smirnov A. (2015, Septiembre) Node.js: ¿Qué es y para qué sirve Node JS? Recuperado el día 23 de octubre de 2015 de <http://www.netconsulting.es/blog/nodejs/>

Solo Electrónicos (2015, Octubre) ¿Qué es y para qué sirve node.js? Recuperado el 23 de octubre de 2015 de <http://soloelectronicos.com/2014/05/27/que-es-y-para-que-sirve-node-js/>

Ubuntu (2015, Octubre) Ubuntu Mate. Recuperado el 23 de octubre de 2015 de https://ubuntu-mate.org/what-is-ubuntu-mate/

Villaba C. & Morales N. (2007) Dispensador de comida para canes de uso doméstico. Tesis de grado universitaria, Universidad de San Buenaventura. Bogotá

WordReference(2005). Dispensar, Recuperado el 15 de octubre del 2015 de <http://www.wordreference.com/definicion/dispensar>

**ANEXOS.**

****Somos estudiantes de la carrera de Ingeniería en Sistemas digitales interactivos que cursan el 3er Semestre. Estamos realizando una investigación titulada: Raspberry Pi, dispensador de comida para mascota y solicitamos de su colaboración para reunir información a través de esta encuesta.

Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Fecha:\_\_\_\_\_\_\_

1. ¿Usted tiene mascotas o ha tenido mascotas?

A) Si B) No

1. ¿Con que frecuencia sale de viaje?

A) Una vez por semana B) Ocasionalmente

C) Una vez al año D) Cada mes

1. Cuando sale de viaje, ¿Se le complica dejar solo a la mascota?

A) Si B) No

1. ¿Tiene acceso de internet en su hogar?

A) Si B) No

1. ¿Le gustaría poder alimentar a su mascota estando fuera del hogar?

A) Si B) No

1. ¿Ha olvidado alguna vez darle de comida a su mascota?

A) Si B) No

1. ¿Le resulta interesante la idea de implementar un dispositivo que le proporcione comida a la mascota que funcione a través del internet?

A) Si B) No

1. ¿Había escuchado alguna vez esta idea?

A) Si B) No

1. ¿Cuánto estaría dispuesto a pagar por un dispositivo así?

Respuesta\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. ¿Este dispositivo le resolvería los problemas de alimentar a su mascota?

A) Si B) No

**RESULTADOS DE LAS ENCUESTAS**

El 90% de las personas han tenido o tiene mascotas.

El 70% de las personas encuestadas viaja ocasionalmente.

El 80% de las personas se le complica dejar sola a la mascota

El 100% de las personas encuestadas tiene internet en su hogar

El 90% de las personas le gustaría alimentar a su mascota estando fuera del hogar.

El 60 % de las personas ha olvidado alguna vez darle comida a su mascota.

El 90% de las personas le resulta interesante la idea del dispositivo.

El 90 % de las personas no habían escuchado alguna vez esta idea

El 60% de las personas pagaría más de mil pesos por el dispositivo

El 90 % de las personas han olvidado alimentar a su mascota por lo menos alguna vez.