UNIVERSIDAD DE TARAPACÁ
Universidad del Estado

Ingeniería
Computación e Informática

SISTEMA DE MONITOREO, CONTROL DE ALIMENTADOR Y BEBEDERO "SMART FEEDER"

Integrantes:
René Ayca
Claudio Carvajal
Yazuska Castillo
Israel Tebes

Asignatura:
Proyecto II

Profesor:
Diego Aracena

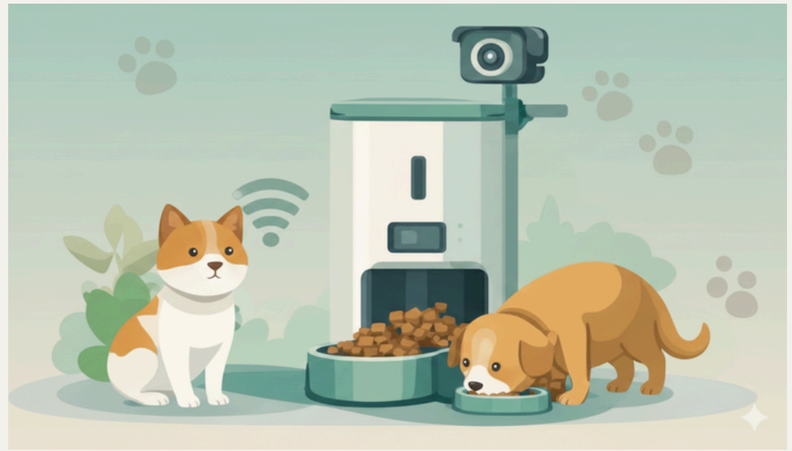
ÍNDICE

INTRODUCCIÓN

La creciente cantidad de personas con mascotas genera el problema de asegurar su alimentación durante las ausencias del dueño. Para solucionarlo, se propone un Dispensador Automático IoT que, mediante una Raspberry Pi 4 Model B y sensores, dispensa comida o agua con precisión. Y Una aplicación móvil que permite programar horarios y supervisar el sistema a distancia. Así, se automatiza la alimentación y se mejora el bienestar de la mascota mediante un prototipo sustentable hecho con materiales reciclados.

PROBLEMA

La alimentación de las mascotas durante la ausencia de sus dueños suele realizarse de forma manual e imprecisa, lo que puede generar horarios irregulares, raciones incorrectas y falta de supervisión, afectando su bienestar y salud.



SOLUCIÓN

Se propone un sistema IoT llamado “Smart Feeder”, que automatiza la alimentación e hidratación de las mascotas. El sistema permite monitorear y programar la alimentación desde una app móvil, garantizando el bienestar y cuidado de las mascotas aun en ausencia de sus dueños.

OBJETIVOS

General

Desarrollar un dispensador automático IoT que se encargue de la alimentación e hidratación precisa para las mascotas. Permitiendo el control y monitoreo remoto por parte del dueño, con el fin de garantizar el bienestar del animal durante largas ausencias.

Específicos

- Adquirir los conocimientos necesarios para el desarrollo del sistema.
- Diseñar maqueta definiendo ubicación de los componentes utilizados para el funcionamiento del dispensador.
- Planificar el desarrollo del proyecto para un avance eficiente.
- Desarrollar una aplicación móvil que permita la conexión con el sistema.
- Realizar pruebas para asegurar que el monitoreo y control automatizado funcione correctamente.
- Analizar y documentar el desarrollo, resultados y conclusiones del proyecto realizado.

PLAN DE INTEGRACIÓN

Integración de Hardware y Software	• Sensores • Actuadores • Raspberry
Integración de la interfaz de usuario	• Diseño y Pruebas • Pruebas de Conectividad
Pruebas del sistema completo	• Simulación de Dispensación • Vista de dispensadores

MODELO DE IMPLEMENTACIÓN

Fase 1: Instalación Física

- Diseño de maqueta y ubicación de sensores y actuadores.
- Conexión de actuadores junto al protoboard.
- Montaje del hardware junto a la maqueta.

Fase 2: Configuración Inicial

- Programación para el funcionamiento de sensores y actuadores.
- Ajuste de parámetros para la dispensación automática.

Fase 3: Pruebas Funcionales

- Simular escenarios junto a la app móvil, sobre la alimentación para evaluar la respuesta del sistema.
- Validar que las alertas y notificaciones funcionen correctamente.
- Capacitación del Usuario Final.
- Proveer un manual de usuario detallado y realizar una demostración práctica de las funciones principales.

MÓDULOS IMPLEMENTADOS

Interfaz de usuario	La interfaz se desarrolló en React Native. Esta permite visualizar la estado de los dispensadores, establecer tiempos para dispensar comida y agua manualmente, visualizar a través de la cámara y programar horarios de alimentación automática.
Controlador central	Se implementó diversas bibliotecas, la cual controla la lógica del sistema haciendo funcionar los sensores, activando relay, actuadores, entre otros.
Sensores y actuadores	Los sensores recogen información sobre el estado de los dispensadores y a la mascota alimentándose. Los actuadores distribuyen alimentos por el tiempo establecido por el usuario.

IMPLEMENTACIÓN DE CÓDIGO

CODIGO.PY > ...

```python
from flask import Flask, jsonify, request, Response
import grovepi
import time
import threading
import json
import datetime
import os
import cv2

app = Flask(__name__)

@app.after_request
def after_request(response):
    response.headers.add('Access-Control-Allow-Origin', '*')
    response.headers.add('Access-Control-Allow-Headers', 'Content-Type')
    response.headers.add('Access-Control-Allow-Methods', 'GET,POST,OPTIONS'
    return response

ULTRA_COMIDA = 4
ULTRA_AGUA = 3
```

```python
def activar_relay(tipo, segundos):
    if tipo == "agua":
        grovepi.digitalWrite(RELAY_AGUA, 1)
        time.sleep(segundos)
        grovepi.digitalWrite(RELAY_AGUA, 0)
```

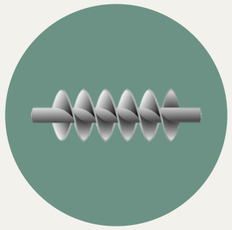
```python
def cargar_config():
    try:
        with open(CONFIG_FILE, "r") as f:
            return json.load(f)
    except:
        return {"horarios": []}
```

PROBLEMAS ENCONTRADOS

Regulador de voltaje

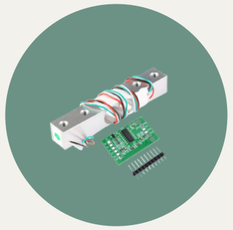
Se tenia una fuente de 12 volts para alimentar a una válvula solenoide y un motor de 3 volts.

Tornillo sin fin



Con un motor de solo 3 volts era incapaz de mover un tornillo sin fin que tenia comida.

Sensor de peso



A pesar de muchos intentos y programación el sensor de peso no daba correcto pesaje, para posteriormente dejar de dar señal.

SOLUCIONES PROPUESTAS

Reemplazo de fuente

Se reemplazo la fuente para alimentar el motor de 3 volts.

Reemplazo de motor

Finalmente a pesar de alimentar el motor de 3 volts, no daba energía suficiente para el tornillo sin fin, por eso se reemplazo por un motor de 12 volts.

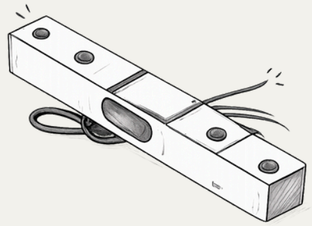
Sensor de peso

Se intento durante un largo periodo de tiempo, pero se optó por quitar el sensor de peso e incluirla en un futuro.

TRABAJO FUTURO

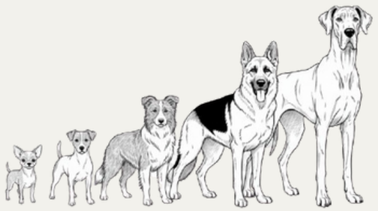
Sensores de peso

Funcionalidad correcta de los sensores



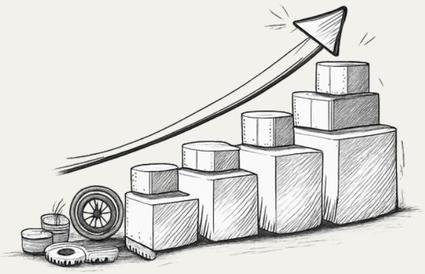
Tamaño personalizado

Un tamaño adecuado para diferentes tipos de mascotas



Multi-Dispensadores

Usar múltiples dispensadores conectados a un solo usuario.



DEMOSTRACIÓN

CONCLUSIÓN

La implementación de un sistema IoT para la automatización del cuidado de mascotas representa una solución innovadora para un problema cotidiano creciente. A lo largo de esta última fase se logró demostrar como fue el análisis, desarrollo e implementación del proyecto.

Los diseños bases de fases anteriores, tanto como la interfaz de usuario y la maqueta, sirvió para una mejor fluides con las funcionalidades del "Smart Feeder". El empleo de un modelo de proceso estructurado y la selección de herramientas y técnicas apropiadas han asegurado que el diseño no solo sea conceptualmente sólido, sino también práctico para un posible trabajo futuro.

BIBLIOGRAFÍA

- Talent.com (n.d). Salario de Jefe de proyecto en Chile. Talent.com. https://cl.talent.com/salary?job=jefe+de+proyecto
- Talent.com (n.d). Salario de Programador en Chile. Talent.com. https://cl.talent.com/salary?job=Programador
- Talent.com (n.d). Salario de Diseñador en Chile. Talent.com. https://cl.talent.com/salary?job=diseñador
- Raspberry Pi y el IoT: guía para entender su papel en el Internet de las Cosas https://monraspberry.com/es/guia-de-raspberry-pi-iot/
- Seeed Studio (n.d). Grove – ADC for Load Cell (HX711). Seeed Studio. https://www.seeedstudio.com/Grove-ADC-for-Load-Cell-HX711-p-4361.html
- Made-in-China (n.d). Stainless Steel Screw Flight Spiral Blade Helical Blade Shaftless for Drilling Machine. https://es.made-in-china.com/co_seitotech/product_Stainless-Steel-Screw-Flight-Spiral-Blade-Helic al-Blade-Shaftless-for-Drilling-Machine_yssisygiuy.html
- Dexter Industries (n.d). GrovePi. https://www.dexterindustries.com/grovepi/

MUCHAS GRACIAS