SMART PET FEEDER

Guided By :
DR.K.LALITHA

PRESENTED By :
FELLAH.A(22AI014)
OVIYASHRI.K(22AI032)
PRADEEPA.P.S.R(22AI035)

TEAM NO: 12

22AIP10 - INTERNET OF THINGS AND ITS APPLICATION LABORATORY

Abstract

- Automated pet feeding system
- Utilizes sensors, motors, and a programmable control unit
- Dispenses food at scheduled intervals
- Portion control mechanisms for effective diet management
- Manual override option for immediate feeding
- Real-time alerts for low food levels
- Easy refilling mechanism for convenience
- Designed to simplify pet care, especially when owners are away
- Aims to provide a reliable, efficient, and user-friendly solution

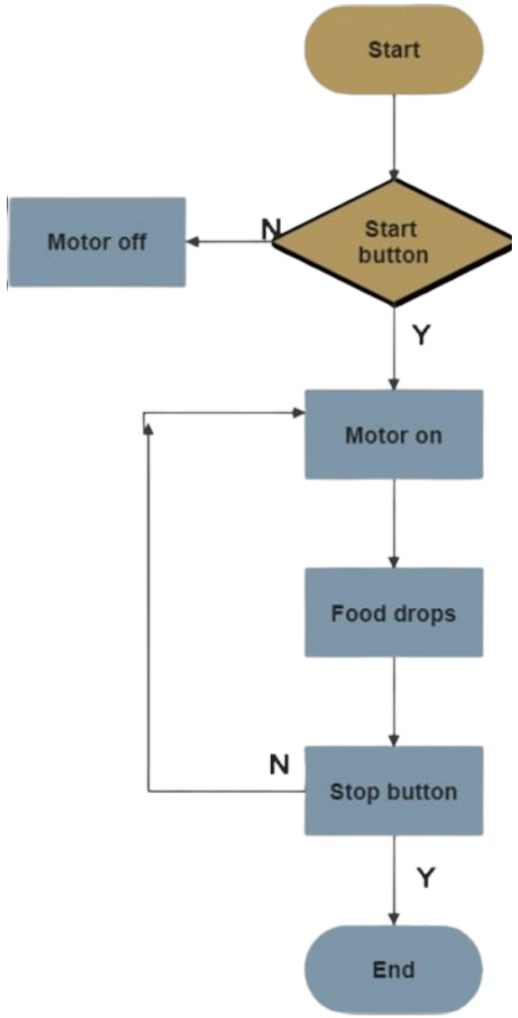
Objectives:

- To create a system that dispenses food automatically at scheduled intervals.
- To ensure accurate and consistent portion sizes to maintain the pet’s health and dietary needs.
- To provide notifications for low food levels or system malfunctions through alerts.
- To ensure the feeder operates reliably and safely, with mechanisms to prevent overfeeding or malfunction.
- To make the system versatile to accommodate different pet types, diets, and feeding schedules.

Components Required:

1. Node MCU
2. Speaker
3. Pet dispenser
4. Servo motor
5. Smartphone with Bluetooth and wifi connection
6. Tuya IOT platform
7. Weight detector

Flow Diagram:



Implementation:
(Hardware or software)

HARDWARE :

- Microcontroller (e.g., Arduino, Raspberry Pi) – to control the system.
- DC Motor/Servo Motor – for dispensing food.
- Motor Driver – to control the motors.
- Load Cell/Weight Sensor – to monitor the portion size.
- Infrared or Ultrasonic Sensor – to detect food level and trigger refills.
- Hopper/Feeding Container – for storing pet food.
- Real-Time Clock (RTC) Module – to schedule feeding times.
- Push Buttons – for manual overrides or setting adjustments.
- Power Supply – to run the entire system.

Implementation:
(Hardware or software)

SOFTWARE :

- Microcontroller Programming Environment (Raspberry Pi) – to write and upload code.
- Embedded C/C++ or Python Code – for motor control, sensor integration, and timing mechanisms.
- Mobile App or Web Interface – for remote monitoring and control.
- Cloud Services (optional) – for real-time notifications and alerts.
- Food Dispensing Algorithm – to manage portion sizes and feeding intervals.
- Sensor Integration Code – to handle input from weight and food level sensors.

Thank u slide