

Pet feeder

Introducere

Pet feeder este un dispozitiv inteligent de hranire pentru pisici, controlabil manual sau prin Wi-Fi. Scopul este de a facilita ingrijirea animalelor de companie, mai ales in lipsa stapanului. Ideea a pornit din nevoia de autonomie si siguranta, util tuturor proprietarilor ocupati.

Descriere generală

Proiectul reprezinta un hranitor automat pentru pisici, bazat pe microcontrolerul ESP32. Sistemul foloseste doua servomotoare SG90 pentru deschiderea compartimentelor de apa si mancare, controlate fie prin doua butoane fizice, fie printr-o interfata web accesibila prin WiFi. Utilizatorul poate interactiona local sau de la distanta, folosind aplicatia web. Un buzzer conectat la un pin GPIO emite un semnal sonor la fiecare actiune, iar un ecran LCD cu interfata I2C afiseaza mesaje precum "Eat well, my kittens" sau "Drink well, my kittens", oferind feedback vizual.



Hardware Design

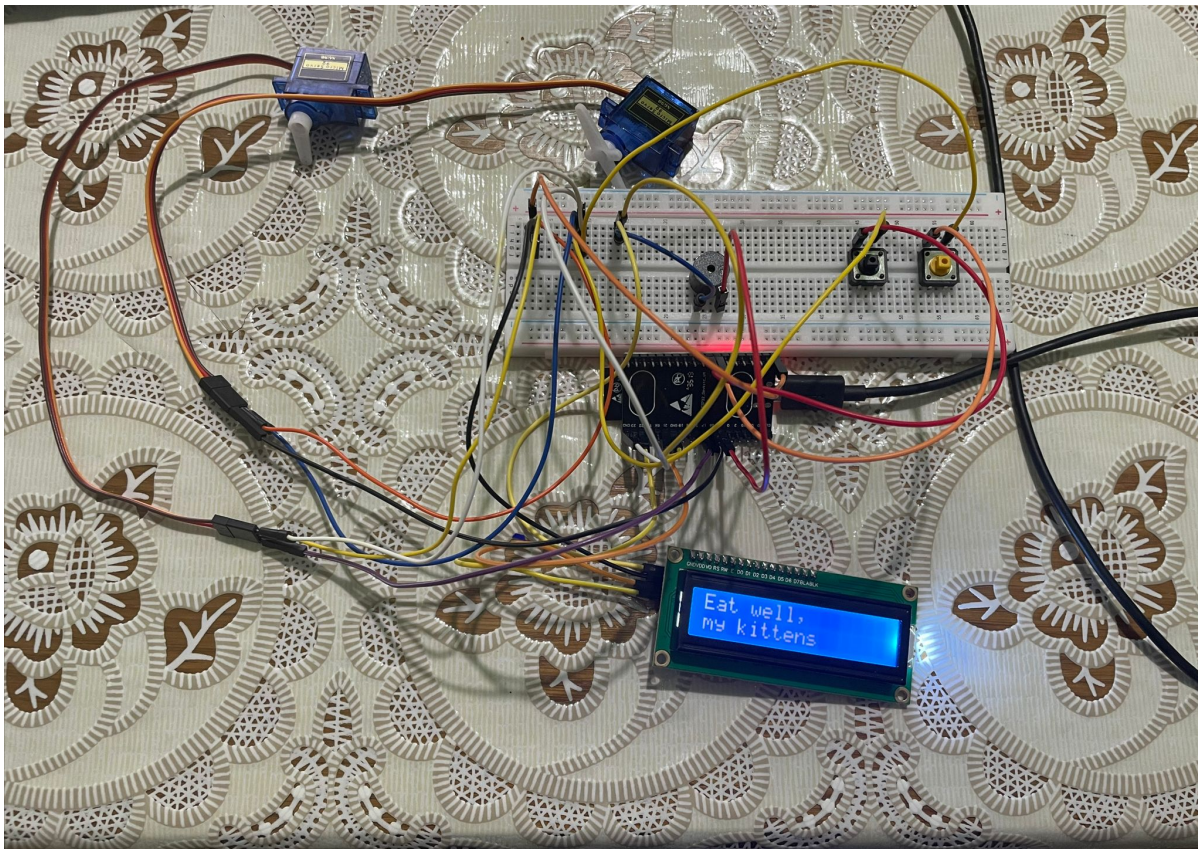
Lista de piese folosite

- [ESP32 Development Board](#)
- [MicroServo SG90](#)
- [Buzzer activ 5V](#)
- [2 Butoane tip push](#)
- [Cabluri mama-tata, mama-mama, tata-tata](#)
- [Breadboard](#)
- [LCD 1602 cu modul I2C](#)
- [Cablu USB micro](#)

Descriere detaliata a functionalitatii hardware



Proiectul este bazat pe placa ESP32, care gestioneaza toate componentele si ofera conexiune WiFi. Doua servomotoare SG90 sunt folosite pentru a actiona mecanismele de eliberare a hranei si apei, fiind conectate pe pinii GPIO16 si GPIO17, controlati prin semnal PWM. Pentru comenzi manuale, sunt conectate doua butoane pe GPIO27 si GPIO14. Acestia sunt setati ca `INPUT_PULLUP`, deci nu au nevoie de rezistori externi. Cand un buton este apasat, ESP32 detecteaza nivelul LOW si executa actiunea corespunzatoare. Pe pinul GPIO4 este conectat un buzzer activ, care emite un semnal sonor scurt la fiecare alimentare. LCD-ul 1602 este conectat prin I2C, folosind pinii SDA (GPIO21) si SCL (GPIO22). Acesta afiseaza mesaje diferite in functie de comanda executata. Toate componentele sunt montate pe breadboard si alimentate printr-un cablu micro USB conectat la calculator. ESP32 alimenteaza si servomotoarele si display-ul, iar consumul total este potrivit pentru o alimentare pe USB, fara sursa externa.



Software Design

- Link Github:

<https://github.com/TeodoraTeo05/cat-feeder-esp32>

- Link Demo video:

<https://youtu.be/CCG9btgEeA4>

Descrierea codului aplicatiei:

- Mediu de dezvoltare: proiectul a fost implementat si testat folosind Arduino IDE, cu placa ESP32 configurata prin Board Manager. Codul a fost scris in C++ si incarcat direct prin cablu USB.
- Librarii si surse 3rd-party utilizate:
 1. WiFi.h - pentru crearea unei retele Wi-Fi in mod Access Point (ESP32 functioneaza ca hotspot local).
 2. WebServer.h - pentru gestionarea cererilor HTTP si afisarea unei pagini HTML de control in browser.
 3. ESP32Servo.h - pentru controlul servomotoarelor SG90, oferind semnal PWM stabil.
 4. LiquidCrystal_I2C.h - pentru afisaj LCD 1602 cu interfata I2C.
- Algoritmi si structuri implementate:
 1. giveFood() - activeaza servomotorul de mancare, afiseaza un mesaj pe LCD si porneste buzzerul.
 2. giveWater() - activeaza servomotorul pentru apa, afiseaza un mesaj relevant pe LCD si activeaza buzzerul.
 3. beepBuzzer() - produce un sunet scurt pentru confirmarea comenzii.
 4. handleRoot() - serveste pagina HTML principala catre browserul utilizatorului.
 5. handleGiveFood() si handleGiveWater() - trateaza cererile GET de la butoanele web si apeleaza functiile corespunzatoare.
- In loop(), se foloseste o structura de polling pentru a verifica starea butoanelor fizice, iar la apasare se declanseaza functiile corespunzatoare (mancare sau apa).
- Interfata HTML include doua butoane care trimit cereri HTTP catre rutele /food si /water, iar serverul ESP32 le gestioneaza in timp real.
- Surse si functii implementate:
 1. setup() initializeaza toate componentele: LCD-ul, servomotoarele, butoanele, buzzerul si serverul web. Creeaza reseaua Wi-Fi cu SSID-ul `CatFeeder_AP`.
 2. loop() verifica permanent daca au fost apasate butoanele fizice si trateaza cererile web de la clientii conectati la retea.
 3. Servomotoarele sunt comandate pe pinii 16 (mancare) si 17 (apa).
 4. Butoanele fizice sunt conectate pe pinii 27 si 14 si sunt citite cu `digitalRead()` in combinatie cu `INPUT_PULLUP`.
 5. Buzzerul este conectat la pinul 4 si este controlat simplu cu `digitalWrite()`.
 6. Pe ecranul LCD apar mesajele: `Eat well, my kittens` sau `Drink well, my kittens`, in functie de comanda primita.

Rezultate Obținute

Pet feeder - produs final:

- https://youtube.com/shorts/fqGpPC8dX_Y?feature=share

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2025/aluca/teodora.teodorescu>



Last update: **2025/05/30 08:25**