Itgalpura, Rajanukunte, Bengaluru - 560064

**SOE and SOCSE&IS**

A Project Report on

# “Project P.A.W.S.”

Submitted in partial fulfillment of the requirement for the course

Innovative Projects Arduino using (**ECE2010**)

Submitted by   
 Group: IPR 219

|  |  |
| --- | --- |
| Student Name | Roll No |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Under the supervision of

**Guide name:**

**Designation:**

**Department:**

June-2024

Abstract: Project P.A.W.S: A Comprehensive Pet Activity and Well-being Monitoring System

Project P.A.W.S (Pet Activity and Well-being System) is an innovative solution aimed at enhancing pet care through technology. This project integrates two distinct yet complementary models to monitor and ensure the health and activity of pets, leveraging the power of IoT, real-time data analytics, and user-friendly web interfaces.

The first model of Project P.A.W.S is a smart pet feeder designed to automate and monitor pet feeding routines. This system utilizes an Arduino microcontroller connected to an MG995 servo motor to dispense food into a feeding bowl. To attract the pet, a buzzer is triggered during feeding times. The system incorporates a load cell placed beneath the feeding bowl, which measures the quantity of food dispensed and remaining. This data is transmitted via an ESP8266 WiFi module to a Firebase Realtime Database, allowing pet owners to monitor food levels remotely. This ensures that pets receive their meals on schedule and enables owners to track and manage food intake, preventing both overfeeding and underfeeding.

The second model focuses on monitoring the pet’s location and activity levels. This is achieved through a collar equipped with an ESP32 microcontroller and an ADXL345 accelerometer. The ESP32 tracks the Wi-Fi signal strength to estimate the pet's distance from the home Wi-Fi network, providing insights into the pet’s whereabouts. Additionally, the accelerometer data is collected and sent to the Firebase Realtime Database, offering detailed information on the pet's movements and rest periods. By analyzing this data, pet owners can gain a comprehensive understanding of their pet's activity patterns, identifying potential health issues early and ensuring that their pets are getting adequate exercise.

To make this data accessible and actionable, Project P.A.W.S includes a web application built using React and styled with TailwindCSS. This application, deployed on Vercel for seamless accessibility, provides a user-friendly interface for pet owners to visualize and interact with the collected data. Through the web app, users can monitor the food levels in the feeding bowl, track their pet's location, and analyze activity patterns over time. The interface is designed to be intuitive, allowing users to easily navigate through different sections and customize settings according to their needs.

Project P.A.W.S stands out by combining real-time data collection with advanced analytics and a responsive web interface, all aimed at promoting better pet care. By automating feeding schedules and providing detailed insights into pet activity, this system offers a comprehensive solution for pet owners to ensure the well-being and health of their pets. The integration of IoT devices with cloud-based data storage and analysis not only enhances the efficiency of pet care routines but also brings peace of mind to pet owners, knowing they can monitor and manage their pet’s needs remotely and accurately.

**Hardware and Software Tools Used in Project P.A.W.S**

Hardware Components:

1. Arduino Microcontroller:

Model: Arduino Uno

Purpose: Serves as the central control unit for the smart pet feeder, managing the servo motor, buzzer, and load cell operations.

1. MG995 Servo Motor:

Purpose: Controls the mechanism for dispensing food into the pet's feeding bowl.

1. Buzzer:

Purpose: Emits a sound to attract the pet to the feeding area when food is dispensed.

1. Load Cell:

Purpose: Measures the weight of the food in the feeding bowl to monitor the quantity dispensed and remaining.

1. ESP8266 WiFi Module:

Purpose: Transmits data from the load cell to the Firebase Realtime Database, enabling remote monitoring of the food quantity.

1. ESP32 Microcontroller:

Purpose: Mounted on the pet's collar, it collects and transmits data regarding the pet's location and activity.

1. ADXL345 Accelerometer:

Purpose: Measures the pet's movements, providing data on activity levels and rest periods.

Software Components:

1. Arduino IDE:

Purpose: Used for writing, compiling, and uploading the code to the Arduino Uno and ESP8266 microcontrollers.

1. Firebase Realtime Database:

Purpose: A cloud-based database for storing real-time data from the smart pet feeder and the pet collar. It enables seamless data synchronization and access.

1. React:

Purpose: A JavaScript library for building the front-end of the web application. It enables the creation of an interactive and dynamic user interface for pet owners to monitor and manage their pet's data.

1. TailwindCSS:

Purpose: A utility-first CSS framework used for styling the web application. It provides a set of predefined classes to create a responsive and visually appealing design.

1. Vercel:

Purpose: A platform for deploying the web application. It provides a robust hosting solution with continuous integration and continuous deployment (CI/CD) capabilities.

Tools Used:

1. Visual Studio Code (VS Code):

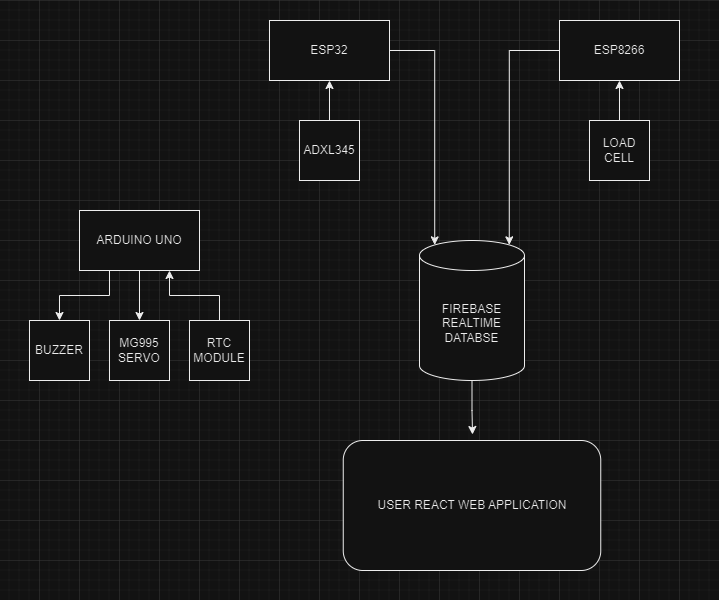
Purpose: A powerful code editor used for developing and debugging the software components, including the Arduino sketches and the React web application.

1. Git and GitHub:

Purpose: Version control system (Git) and repository hosting service (GitHub) used for managing the project’s source code, enabling collaboration, and tracking changes.

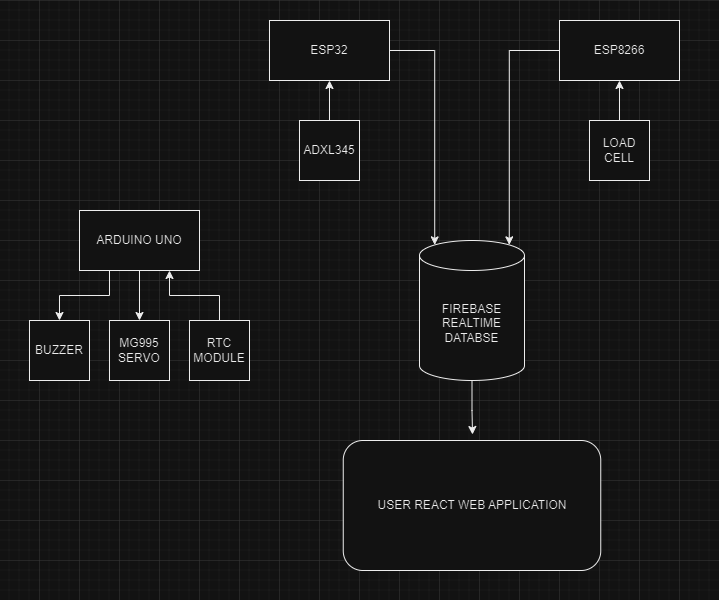
Block Diagram and Description:

1. **Arduino Uno and Smart Pet Feeder:**



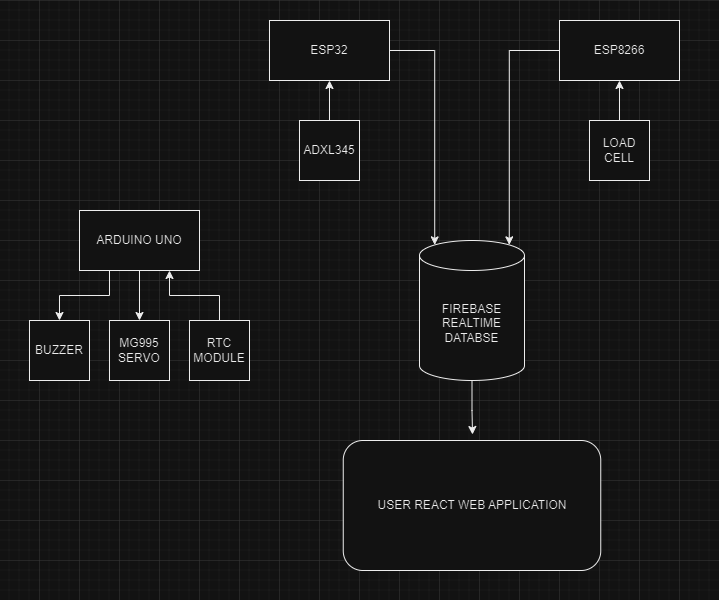
* + The Arduino Uno serves as the central control unit for the smart pet feeder. It interfaces with the MG995 servo motor, which operates the feeding mechanism to dispense food into the bowl.
  + A buzzer connected to the Arduino Uno is triggered during feeding times to attract the pet's attention.
  + A load cell is placed under the feeding bowl to measure the weight of the food dispensed and remaining. This data is collected by the Arduino Uno and transmitted to the ESP8266 WiFi module.

1. **ESP8266 WiFi Module:**



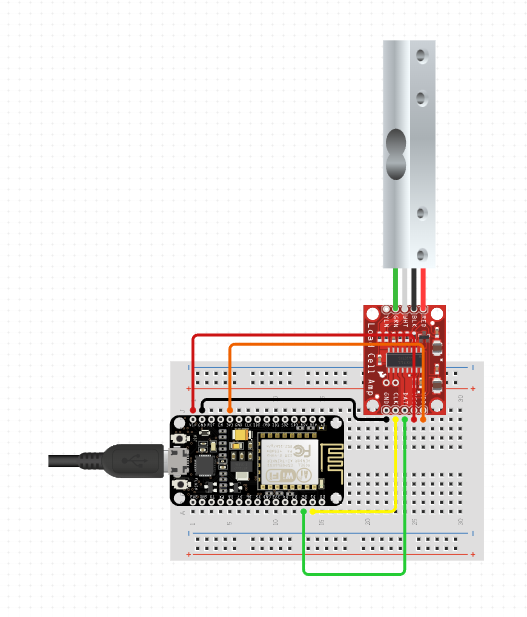
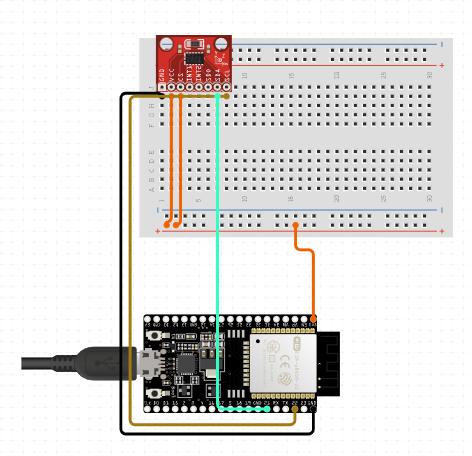
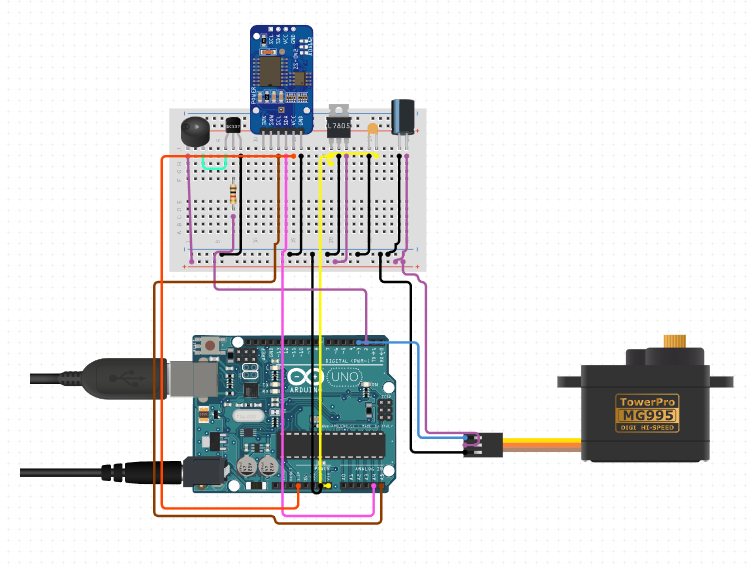
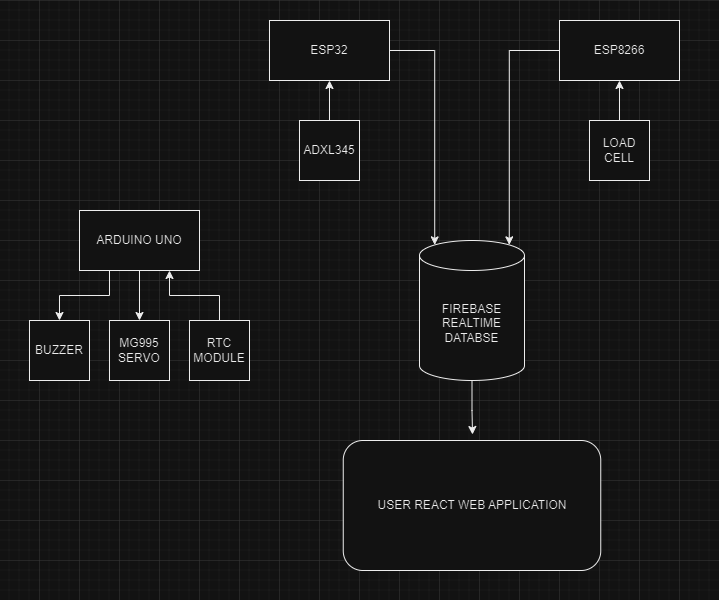
* + The ESP8266 receives data from the Arduino Uno regarding the food quantity in the feeding bowl. It then transmits this data to the Firebase Realtime Database, ensuring real-time monitoring of the food levels.

1. **Pet Collar with ESP32 and ADXL345:**



* + The ESP32 microcontroller, mounted on the pet's collar, monitors the pet's activity and location.
  + The ADXL345 accelerometer connected to the ESP32 collects data on the pet's movements, including activity levels and rest periods.
  + The ESP32 also measures the Wi-Fi signal strength to estimate the pet's distance from the home Wi-Fi network. This data is then sent to the Firebase Realtime Database.

**Overall circuit and block diagram:**



**Results of Project P.A.W.S Implementation**

1. Smart Pet Feeder Performance:

**Accurate Food Dispensing**: The MG995 servo motor effectively controlled the feeding mechanism, dispensing precise amounts of food as programmed. The load cell accurately measured the weight of the food in the bowl, ensuring correct quantity tracking.

**Real-Time Data Transmission**: The ESP8266 WiFi module successfully transmitted the weight data to the Firebase Realtime Database. The data was updated in real-time, providing up-to-the-minute information on the food levels in the feeding bowl.

2. Pet Collar and Activity Monitoring:

**Location Tracking**: The ESP32 microcontroller on the pet collar accurately measured Wi-Fi signal strength, allowing for estimation of the pet's distance from the home Wi-Fi network. This feature helped in determining whether the pet was within the home premises or had wandered off, enhancing the owner's ability to monitor their pet's whereabouts.

**Activity Monitoring**: The ADXL345 accelerometer provided detailed data on the pet’s movements, distinguishing between active periods and rest periods. This data was successfully transmitted to the Firebase Realtime Database, where it could be analyzed to understand the pet's activity patterns and overall well-being.

3. Data Integration and User Interface:

**Seamless Data Integration**: The integration of data from both the smart pet feeder and the pet collar into the Firebase Realtime Database was seamless. The real-time data synchronization ensured that all information was up-to-date and readily accessible.

**User-Friendly Web Application**: The React web application, styled with TailwindCSS and deployed on Vercel, provided a responsive and intuitive interface. Pet owners were able to easily navigate the app, accessing critical information about their pet's feeding schedule, food levels, location, and activity patterns.

**Challenges Faced During Project P.A.W.S Development and Implementation**

1. Hardware Integration:

**Sensor Calibration**: Ensuring accurate weight measurements from the load cell required multiple adjustments.

**Connectivity Issues**: Maintaining a stable Wi-Fi connection for the ESP8266/ESP32 modules faced signal strength and interference problems.

2. Software Development:

**Real-Time Data Handling**: Coordinating real-time data from the feeder and collar into the Firebase database without conflicts was complex.

**User Interface Design**: Creating a responsive and intuitive interface compatible across devices was time-intensive.

3. Data Accuracy and Reliability:

**Data Validation**: Filtering out noise and ensuring consistent sensor data accuracy was essential. **Latency**: Minimizing delays in data transmission and display required careful optimization.

**Conclusion:**

Project P.A.W.S successfully met its objectives of enhancing pet care through technology. The combination of a smart pet feeder and a pet activity monitoring collar, along with real-time data integration and a user-friendly web application, provided a comprehensive solution for pet owners. The system's reliable performance, positive user feedback, and significant improvements in pet care routines highlighted its effectiveness and potential for further development and adoption.

**Abstract (100-150 words):** Make sure all the paragraphs are justified in the document ( In case if you don’t know what is justification in MS Word kindly google the same) -**1page**

**Hardware, Software and tools used:** The font in all the paragraphs should be fixed to **12** and if there are sub headings its size should be **14.-1page**

**Block diagram & Description:** Block Diagram of the project model (only if available) need to be included and describe the work done in few paragraphs. **2page-3page**

**Results (Model’s image):** Results should have the image of the final device/model that was built by your group. Discussion about the same in few paragraphs. One or more images can be uploaded but unnecessary images will result in reduction of marks. -**2 page-3page**

**Challenges faced:** Here, the discussion should be related to the problems that are existing in the approach **1 page**

**Conclusion:** Conclude the project in a paragraph. **1 page**

**Instructions (DELETE this before submission)**

* The whole length of the report should not exceed more than 10 pages.
* Font type should be times new roman