BAB III
METODE PENELITIAN

3.1 Alat dan Bahan

3.1.1 Alat

Sebelum memulai *Rancang bangun kontrol monitoring alat pakan dan minum kucing berbasis IoT,* Berikut alat yang akan dipersiapkan dapat dilihat pada Tabel 3.1 dibawah ini.

Tabel 3. 1 Alat yang digunakan

No	Nama Alat	Spesifikasi	Fungsi	Jumlah
1.	*Laptop*	16GB *RAM*/512GB *ROM*	Sebagai Perancangan Dan Pemrograman Sistem	1 Buah
2.	Obeng-Obeng Set	-	Sebagai Membuka Atau Mengencangkan Baut Pada Kompenen	1 Buah
3.	Solder	80 Watt	Sebagai Penyambung Atau Melengkapi Antara Timah Dan Komponen	1 Buah
4.	Arduiuno IDE	Arduiuno 2.03	Sebagai Proses *Uplod* Kode Program Ke Alat Yang Dibua	1 Buah

3.1.2 **Bahan**

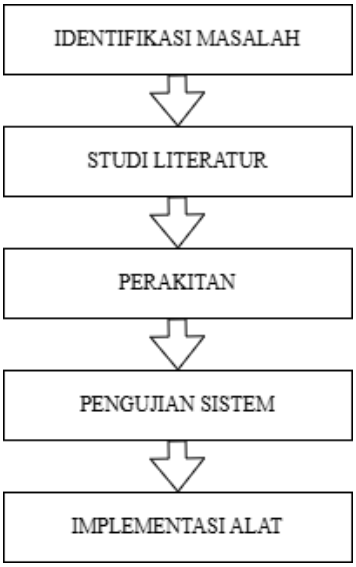
Sebelum memulai proses rancang bangun kontrol monitoring alat pakan dan minum kucing berbasis IoT, Berikut adalah daftar komponen yang perlu disiapkan untuk penelitian ini, yang akan dijelaskan lebih lanjut dalam tabel 3.2

Tabel 3. 2 Bahan

No	Nama Bahan	Spesifikasi	Fungsi	Jumlah
1	ESP32	32-bit LX6 dual-core 240 MHz	Sebagai Wi-Fi, Bluetooth, pemrosesan data, kontrol dispenser, sensor berat, kamera, notifikasi, komunikasi serial, penghematan energi.	1 Buah
3	Motor Servo	-	Berfungsi untuk mengatur porsi makanan, membuka/menutup penutup, dan memastikan distribusi makanan yang tepat.	1 Buah
4	Loadcell Sensor	1Kg	Untuk mengukur berat atau beban hingga 1 kg.	1 Buah
5	Loadcell Sensor	10Kg	Untuk mengukur berat atau beban hingga 5 kg	1 Buah
6	Kabel Micro USB	Panjang 30cm	Sebagai Alat komunikasi Komputer Antar alat elektronik	1 Buah
7	Mini Pompa Air	-	Mengisi wadah air minum kucing secara otomatis saat level air rendah, memastikan kucing selalu memiliki akses ke air bersih.	1 Buah
8	Relay	-	Sebagai pemutus arus Listrik ke mini pompa air	1 Buah

3.2 Tahapan Penelitian

Berikut ini adalah tahapan penelitian yang di gunakan pada penelitian ini dengan digambarkan dalam bentuk blok diagram pada gambar 3.1



Gambar 3. 1 Diagram block tahapan penelitian

5. Identifikasi Masalah

   Langkah awal yang sangat krusial dalam proses penelitian ialah mengenali permasalahan yang akan diselidiki. Langkah ini dapat dikerjakan dengan memeriksa masalah yang hendak diselidiki. Setelah itu, diambil langkah-langkah untuk memahami lebih mendalam, baik dengan melakukan pengamatan maupun membaca literatur terkait.

6. Studi Literatur

   Dalam pendekatan ini, penulis melakukan pencarian bahan literatur untuk penulisan skripsi melalui berbagai pustaka dalam penelitian terkait, fokus pencarian adalah terkait dengan *Rancang bangun kontrol monitoring alat pakan dan minum kucing berbasis IoT.*

7. Perancangan Sistem

   Sistem alat pakan dan minum kucing otomatis memberikan makan dengan Motor servo mengeluarkan pakan, sementara mini pompa air mengisi wadah minum saat pengguna menekan tombol on pada aplikasi. Sensor load cell 10 Kg memantau berat kucing dan motor servo akan membuka sesuai berat yang di baca oleh load cell 10 Kg,dan load cell 1 Kg memantau berat dari pakan yang

di dalam wadah dan data dikirim ke aplikasi di smartphone. Dengan sistem ini, perawatan kucing menjadi lebih mudah.

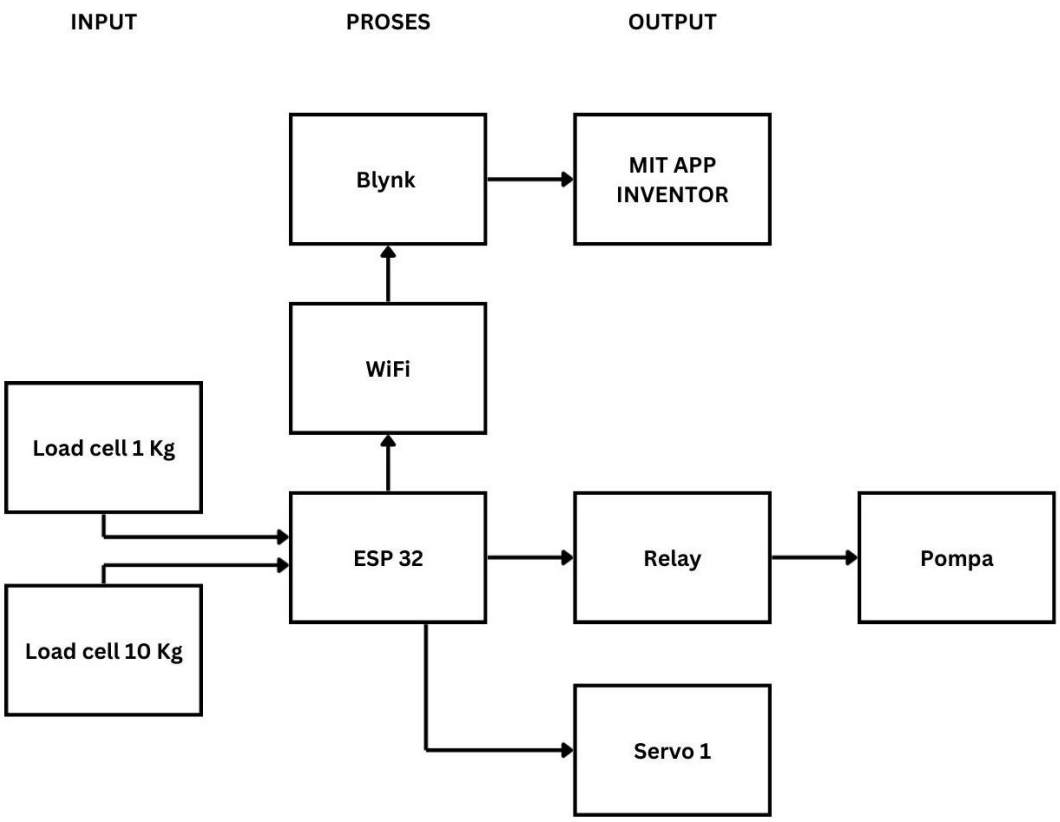
8. Pengujian Sistem

   Proses uji coba sistem alat pakan kucing dilakukan Memeriksa bahwa perangkat yang telah dibuat beroperasi sesuai dengan rencana perancangan dan untuk mengenali serta menyelesaikan kemungkinan kesalahan yang mungkin terjadi dalam perangkat tersebut.

9. Implementasi alat tempat pakan kucing dimulai dengan memilih komponen seperti ESP32, load cell, motor servo, mini pompa. Setelah merakit perangkat keras, perangkat lunak dikembangkan untuk mengatur pemberian minum kucing dan memonitoring berat pakan dan berat kucing. Semua sistem diintegrasikan dan diuji, memastikan kerja otomatis sempurna. Setelah terpasang di ruangan, pemilik dapat mengontrol dan memantau kucing dari jarak jauh dengan mudah dan praktis.

3.3 Perancangan Perangkat Keras

Merancang sistem dengan tujuan untuk mempermudah proses pembuatan alat Ide dasar dalam merancang Sistem pakan otomatis dan monitoring berat badang kucing serta kamera pengawas. Diilustrasikan melalui sebuah diagram blok yang tersedia pada gambar dibawah ini Diagram blok ini menyajikan gambaran keseluruhan. untuk perancang perangkat keras bisa dilihat pada gambar 3.3.



Gambar 3. 2 Diagram Block

Dengan mengacu pada gambar diagram blok sistem, langkah awalnya ialah :

1. **Load cell 1 Kg**

   Berfungsi untuk mengukur berat pakan yang keluar dari wadah penampungan. Data dari load cell ini digunakan untuk memantau jumlah pakan yang dikeluarkan secara real-time dan ditampilkan di aplikasi. Pada pengaturan sistem ini, load cell 1 Kg hanya mengirimkan data pembacaan dari berat pakan yang di dalam wadah ke ESP 32, sehingga load cell 1 kg berperan utama sebagai alat monitoring.

2. **Load cell 10 Kg**

   Digunakan untuk mengukur berat badan kucing. Hasil pengukuran berat di gunakan sebagai syarat untuk membuka menutup servo pakan. Jika berat badan kucing kurang dari 3 Kg maka servo akan membuka selama 1 detik dan akan

menurunkan pakan sebanyak 24 gram dan jika berat kucing terdeteksi 3 Kg hingga 5,5 Kg makan servo akan membuka selama 1.5 detik dan akan memberikan pakan seberat 33 gram. Data ini dapat membantu pemilik memantau perubahan berat badan kucing dari waktu ke waktu. Pada tahap ini akan di berikan delay agar tidak memberikan pakan kucing secara terus menerus, delay yang saya berikan adalah 1 jam.

3. **ESP 32**

   Menjadi pusat kendali seluruh sistem, yang bertugas membaca data dari kedua load cell, menerima perintah dari aplikasi Blynk atau MIT App Inventor, dan mengontrol komponen output seperti relay dan servo. ESP32 juga mengatur komunikasi data melalui Wi-Fi, memastikan bahwa semua informasi sensor terkirim ke aplikasi dan perintah dari pengguna dapat dijalankan secara real-time.Setelah semua proses selesai, sistem akan kembali dalam kondisi siaga. dan menunggu siklus berikutnya untuk kembali menjalankan proses pemberian makan, minum, dan pemantauan berat badan sesuai jadwal yang telah ditentukan.

4. **Wi-Fi**

   Sebagai penghubung antara ESP32, dan aplikasi pengguna. Melalui jaringan ini, data sensor, dan perintah kontrol dikirim dan diterima secara cepat. Wi-Fi memastikan integrasi antara perangkat keras dan aplikasi berbasis IoT, sehingga sistem dapat diakses dari jarak jauh selama terhubung ke internet. Stabilitas jaringan menjadi faktor penting agar komunikasi berjalan lancar.

5. **Servo**

   Digunakan untuk membuka dan menutup mekanisme drop pakan. Servo ini bergerak ke posisi 90° untuk membuka saluran pakan dan kembali ke 0° untuk menutupnya. Perintah gerak servo diberikan secara otomatis jika load cell membaca berat badan kucing. Jika berat badan kucing kurang dari 3 Kg maka servo akan membuka selama 1 detik dan akan menurunkan pakan sebanyak 24

gram dan jika berat kucing terdeteksi 3 Kg hingga 5,5 Kg makan servo akan membuka selama 1.5 detik dan akan memberikan pakan seberat 33 gram. sedangkan load cell 1 kg hanya digunakan untuk memantau jumlah pakan yang keluar.

6. Relay dan Pompa Air

Berfungsi menghidupkan atau mematikan pompa air yang digunakan untuk menyediakan minuman bagi kucing. Relay dikontrol melalui pin D19 pada ESP32 dan hanya dioperasikan secara manual melalui aplikasi. Sistem ini memungkinkan penghematan energi karena pompa hanya menyala ketika diperlukan.

7. Blynk

Digunakan sebagai platform kontrol jarak jauh dan monitoring data. Melalui dashboard Blynk, pengguna dapat melihat berat pakan, berat kucing, serta mengontrol Servo 1 dan relay pompa. Blynk juga berperan sebagai perantara komunikasi antara ESP32 dan MIT App Inventor.

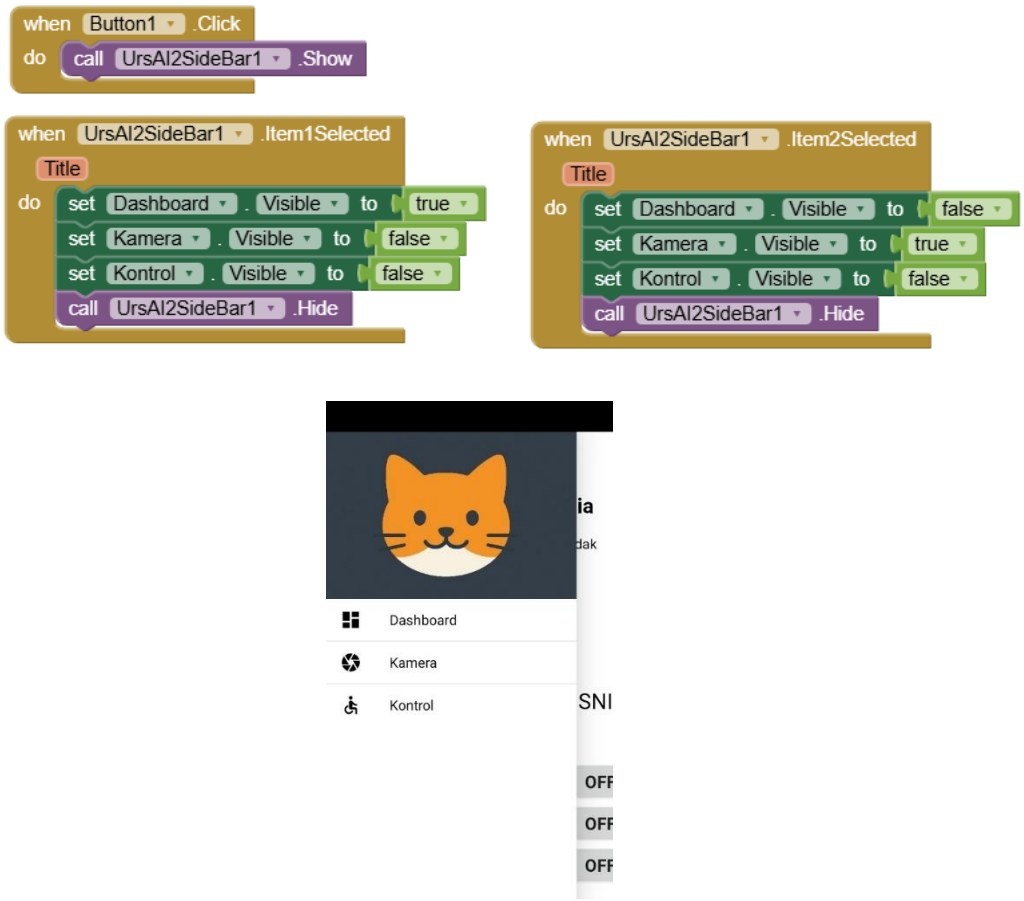
8. MIT App Inventor

digunakan sebagai antarmuka visual yang menyediakan tombol kontrol untuk menggerakkan Servo, serta memungkinkan pengguna memantau data sensor. Integrasi dengan Blynk atau koneksi langsung ke ESP32 membuat aplikasi ini menjadi pusat kendali dan monitoring.

3.4 Rancangan Aplikasi

Pada tahap perancangan perangkat lunak ini, penulis menggunakan MIT App Inventor untuk membuat tampilan aplikasi yang akan digunakan sebagai antarmuka sistem. MIT App Inventor dipilih karena menawarkan kemudahan dalam merancang aplikasi Android melalui metode *block programming*, sehingga proses pembuatan menjadi lebih cepat tanpa memerlukan penulisan kode yang rumit. Melalui aplikasi ini, pengguna dapat dengan mudah

memantau, mengontrol, dan berinteraksi dengan perangkat keras yang telah dibuat. Desain tampilan dibuat sesederhana dan sejelas mungkin agar mudah digunakan oleh siapa saja, sekaligus mendukung proses komunikasi data antara perangkat keras dan aplikasi secara real-time. Tampilan MIT App Inventor dapat dilihat pada gambar di bawah ini



Gambar 3. 3 Contoh Code Block dan tampilan di smart phone

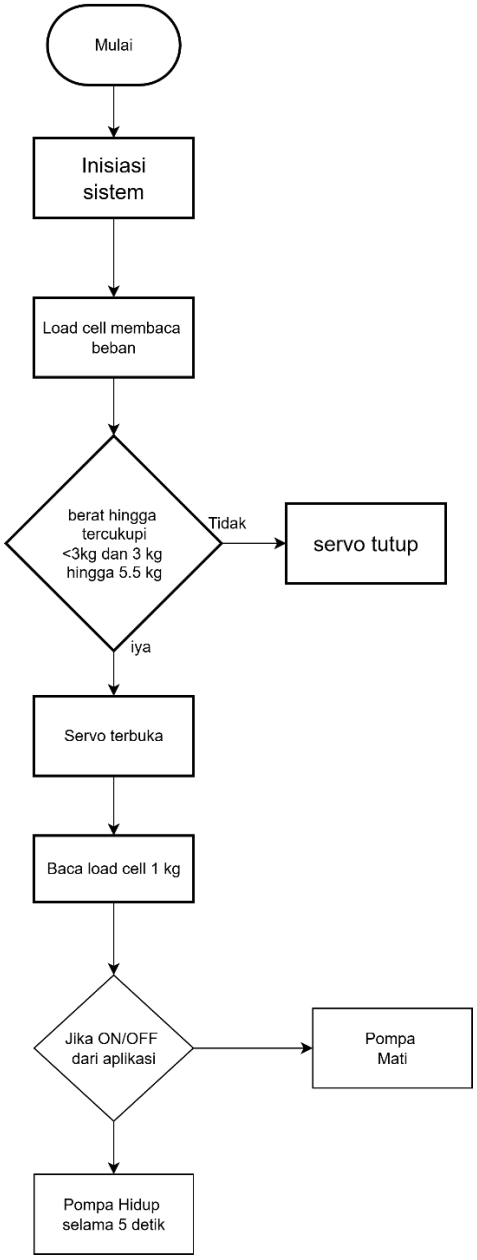
Gambar di atas menunjukkan tampilan *code blocks* pada MIT App Inventor yang digunakan untuk mengatur alur logika dan fungsi dari aplikasi. Setiap blok merepresentasikan perintah atau aksi tertentu yang saling terhubung untuk membentuk rangkaian program yang utuh. Dengan metode ini, proses

pemrograman menjadi lebih visual, mudah dipahami, dan meminimalkan kemungkinan terjadinya kesalahan penulisan kode.

Selain *code blocks*, aplikasi yang telah dibuat pada MIT App Inventor juga dapat dijalankan dan diuji langsung melalui perangkat smartphone Android. Tampilan aplikasi di handphone dirancang agar sederhana, responsif, dan mudah digunakan oleh pengguna. Setiap menu, tombol, dan indikator disusun dengan tata letak yang jelas sehingga memudahkan proses pemantauan dan pengendalian perangkat keras secara real-time. Dengan adanya tampilan ini, pengguna dapat mengoperasikan sistem hanya melalui sentuhan layar, tanpa perlu terhubung langsung ke perangkat keras.

3.5 Perancangan Perangkat Lunak

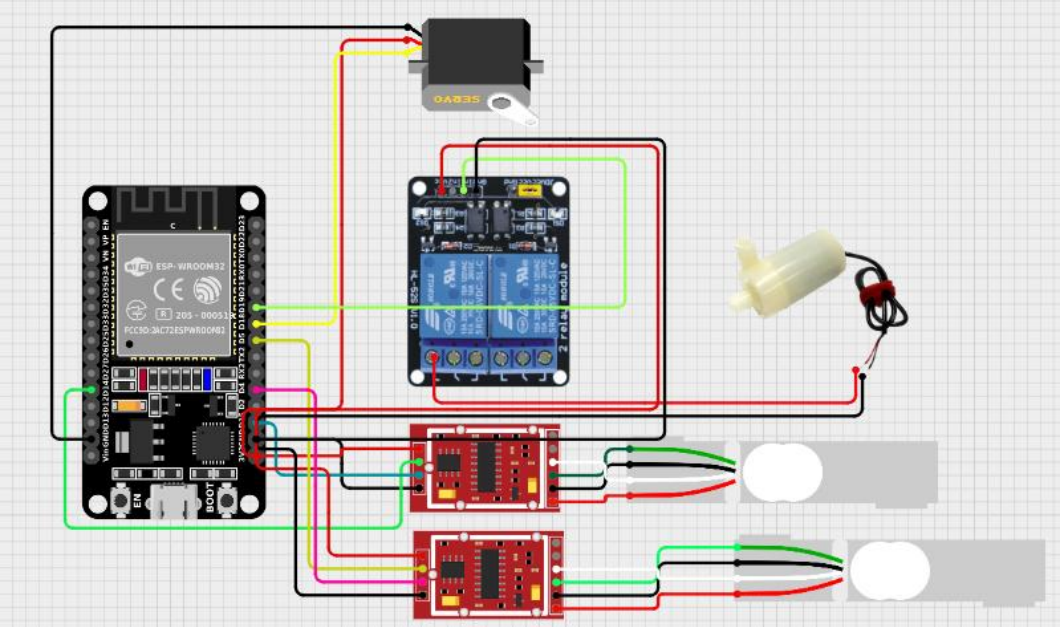
Perancangan perangkat lunak ini meliputi penyusunan diagram alur hingga implementasi sistem pemberian pakan kucing berbasis mikrokontroler ESP32. Yang dapat di lihat pada *flowchart* perancangan perangkat lunak Sistem ini dilengkapi dengan dua load cell, yaitu load cell 1 kg untuk mengukur berat pakan dan load cell 10 kg untuk memantau berat badan kucing. Selain itu, terdapat relay yang berfungsi untuk mengontrol pompa air mini, servo 1 untuk mekanisme penurunan pakan, serta servo 2 yang digunakan sebagai penggerak radar kamera. Perangkat lunak dirancang agar dapat terhubung dengan platform Blynk untuk pengiriman dan penerimaan data secara real-time, sekaligus mengunggah data ke MIT App Inventor melalui koneksi Wi-Fi. Seluruh alur kerja sistem digambarkan pada flowchart



Gambar 3. 4 Flowchart Alat Pakan Kucing

3.6 Perancangan Rangkaian Keseluruhan

Instalasi keseluruhan ini yaitu pada tahap akhir dari perancangan rangkaian yang diimplementasikan, untuk selanjutnya semua komponen dipasang sesuai dengan kebutuhan sistem yang dibuat, untuk instalasi keseluruhannya dapat di lihat pada gambar 3.13



Gambar 3. 5 Rangkaian Alat yang dibuat dengan cirkit dsiginer

3.7 Pengujian Alat

1. Mikro Kontroler ESP 32

Pengujian dilakukan dengan mengunggah program utama ke dalam mikrokontroler ESP32 dan memantau apakah perangkat dapat menjalankan fungsi-fungsi yang telah dirancang, seperti membaca data dari sensor load cell (berat pakan dan berat badan kucing), mengontrol pergerakan servo untuk membuka dan menutup katup pakan, mengaktifkan relay untuk pompa air, serta mengirim dan menerima data melalui koneksi WiFi menggunakan aplikasi Blynk IoT. Hasil pengujian menunjukkan bahwa modul ESP32 dapat bekerja dengan stabil, mampu merespons perintah sesuai logika program, dan dapat berkomunikasi dengan perangkat lunak secara real-time tanpa gangguan.

kucingblynkmit.ino

```
#define BLYNK_TEMPLATE_ID "TMPL6pG5n8HKl"
#define BLYNK_TEMPLATE_NAME "Pakan Kucing Berbasis Iot"
#define BLYNK_AUTH_TOKEN "pXhsW8g9p7FVSfnVnLv2GyW1n3hdU2g1"

#include <HX711_ADC.h>
#include <ESP32Servo.h>
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>

// WiFi
char ssid[] = "realme 7i";
char pass[] = "Yasirli221";
```

Gambar 3. 6 Potongan Code Program ESP 32

2. Load Cell 1 Kg

Pengujian dilakukan dengan menghubungkan sensor load cell 1 kg ke modul HX711 dan mikrokontroler ESP32, kemudian menjalankan program pembacaan berat untuk memastikan sensor dapat mengukur massa pakan di dalam wadah. Proses kalibrasi dilakukan menggunakan beban standar agar hasil pembacaan akurat, dengan fokus pada rentang pengukuran hingga 1 kg. Hasil pengujian menunjukkan bahwa sensor load cell mampu mendeteksi perubahan berat secara real-time dengan tingkat keakuratan yang memadai, serta dapat mengirimkan data berat ke aplikasi Blynk IoT melalui koneksi WiFi. Pengujian juga membuktikan bahwa sensor tetap stabil dalam membaca data meskipun dilakukan pengisian atau pengurangan pakan secara bertahap.

```
// Pin Load Cell
#define DOUT_KUCING 4
#define CLK_KUCING  5
#define DOUT_PAKAN 14
#define CLK_PAKAN  15

// Pin Servo & Relay
#define SERVO_PIN       18   // Servo pakan
#define SERVO_KAMERA_PIN 21  // Servo kamera
#define RELAY_PIN       19   // Pompa

// Objek
HX711_ADC scaleKucing(DOUT_KUCING, CLK_KUCING);
HX711_ADC scalePakan(DOUT_PAKAN, CLK_PAKAN);
Servo servoPakan;
Servo servoKamera;
```

Gambar 3. 7 Potongan code program Load cell 1 Kg

3. Load Cell 10 Kg

Pengujian dilakukan dengan menghubungkan sensor load cell 10 kg ke modul HX711 dan mikrokontroler ESP32, kemudian menjalankan program pembacaan berat untuk memantau massa tubuh kucing yang berada di atas platform timbangan. Proses kalibrasi dilakukan menggunakan beban yang telah di timbang dengan timbangan digital untuk memastikan pembacaan sesuai dengan berat sebenarnya. Hasil pengujian menunjukkan bahwa sensor load cell 10 kg mampu membaca perubahan berat secara akurat dan stabil pada rentang pengukuran yang luas, sehingga dapat memantau berat badan kucing dengan baik. Data hasil pengukuran dapat dikirimkan secara real-time ke aplikasi Blynk IoT melalui koneksi WiFi.

```
// Pin Load Cell
#define DOUT_KUCING 4
#define CLK_KUCING  5
#define DOUT_PAKAN 14
#define CLK_PAKAN  15

// Pin Servo & Relay
#define SERVO_PIN       18   // Servo pakan
#define SERVO_KAMERA_PIN 21  // Servo kamera
#define RELAY_PIN       19   // Pompa

// Objek
HX711_ADC scaleKucing(DOUT_KUCING, CLK_KUCING);
HX711_ADC scalePakan(DOUT_PAKAN, CLK_PAKAN);
Servo servoPakan;
Servo servoKamera;

// Kalibrasi (ubah jika perlu)
float kalibrasiKucing = 220.90;     // Kalibrasi load cell kucing (10kg)
float kalibrasiPakan = 1108.77;     // Kalibrasi load cell pakan (1kg)
```

Gambar 3. 8 Potongan Code Program Load cell 10 Kg

4. SERVO

Pengujian dilakukan dengan menghubungkan servo motor pertama ke mikrokontroler ESP32 yang berfungsi untuk mengatur pembukaan dan penutupan katup pakan. Servo diuji mode otomatis berdasarkan jadwal pemberian makan. Pada mode otomatis, servo bergerak ke sudut 90° jika sensor load cell 10 Kg mendeteksi berat badan kucing kurang dari 3 Kg maka servo akan membuka selama 1 detik dan akan menurunkan pakan sebanyak 24 gram dan jika berat kucing terdeteksi 3 Kg hingga 5,5 Kg maka servo akan membuka selama 1.5 detik dan akan memberikan pakan seberat 33 gram. untuk membuka aliran pakan dan kembali ke sudut 0° untuk menutup servo. Pada mode manual, pergerakan servo dapat dikendalikan langsung dari aplikasi dengan respons waktu yang cepat. Hasil pengujian menunjukkan bahwa servo dapat beroperasi secara stabil, memiliki sudut pergerakan presisi, dan mampu menjalankan perintah sesuai logika program tanpa adanya hambatan mekanis pada sistem pakan.

```
148     if (beratKucing < 1000) {
149       Serial.println("⚠ Berat kucing belum mencapai 1 kg");
150     } else if (now - lastFeedTime <= delayServo) {
151       unsigned long sisa = delayServo - (now - lastFeedTime);
152       Serial.print("⏳ Jeda feeding: ");
153       Serial.print(sisa / 1000); Serial.println(" detik");
154     } else {
155       if (isStable(scaleKucing)) {
156         int waktuBuka = durasiServo(beratKucing);
157         if (waktuBuka > 0) {
158           feeding = true;
159           beratAwalPakan = beratPakan;
160           Serial.println("✅ Kucing stabil! Servo buka pakan.");
161           Serial.print("⏱ Durasi buka servo: "); Serial.print(waktuBuka / 1000.0, 1); Serial.println(" detik");
162 
163           // Servo buka → tunggu → tutup
164           servoPakan.write(90);
165           delay(waktuBuka);
166           servoPakan.write(0);
167         } else {
```

Gambar 3. 9 Code Program Servo Pakan

5. RELAY

Pengujian dilakukan dengan menghubungkan modul relay ke mikrokontroler ESP32 yang berfungsi untuk mengontrol pompa air pada sistem alat pakan kucing berbasis IoT. Pengujian dilakukan dalam mode manual melalui aplikasi Blynk IoT,

di mana pengguna dapat menyalakan secara manual dan pompa akan mati secara otomatis dari perangkat smartphone. Hasil pengujian menunjukkan bahwa relay dapat bekerja dengan baik, mampu menghubungkan dan memutus arus listrik ke pompa air sesuai perintah dari aplikasi dengan respon waktu yang cepat. Selama pengujian, relay beroperasi secara stabil tanpa mengalami panas berlebih atau kegagalan fungsi, sehingga dapat diandalkan untuk mengatur suplai air minum kucing secara aman.

```
// Relay Pompa
pinMode(RELAY_PIN, OUTPUT);
digitalWrite(RELAY_PIN, HIGH);  // Mati saat awal
statusPompa = "OFF";

Serial.println(">>> Sistem siap.\n");
```

Gambar 3. 10 Code Program Relay

6. POMPA

Pengujian dilakukan dengan menghubungkan pompa air ke modul relay yang dikendalikan oleh mikrokontroler ESP32. Tujuan pengujian ini adalah memastikan pompa dapat bekerja sesuai perintah yang dikirim melalui aplikasi Blynk IoT. Proses pengujian dilakukan dengan menyalakan pompa secara manual dan akan mati secara otomatis dari aplikasi, Hasil pengujian menunjukkan bahwa pompa dapat menyala dan berhenti dengan cepat sesuai perintah, serta mampu mengalirkan air secara stabil tanpa hambatan pada selang maupun sumbatan pada saluran keluaran. Selama pengujian, pompa bekerja dengan suhu operasional yang normal dan tidak menimbulkan getaran atau kebisingan berlebihan, sehingga aman digunakan untuk sistem pengisian minum kucing.

```
BLYNK_WRITE(V0) {
  int pinValue = param.asInt();
  if (pinValue == 1) {
    digitalWrite(RELAY_PIN, LOW);   // Aktifkan pompa
    Serial.println("💧 Pompa ON (otomatis mati 5 detik)");
    delay(5000);
    digitalWrite(RELAY_PIN, HIGH);  // Matikan pompa
    Blynk.virtualWrite(V0, 0);      // Reset tombol di Blynk
    Serial.println("💧 Pompa OFF otomatis");
  }
```

Gambar 3. 11 Code Program Pompa Mini yang terhubung dengan relay