# LAMPIRAN

## Lampiran 1 Surat Kesediaan Pembimbing 1

**SURAT KESEDIAAN MEMBIMBING TA**

Yang bertanda tangan di bawah ini:

| | |
|---|---|
| Nama | : Safar Dwi Kurniawan, M.Kom |
| NIPY | : 03.021.487 |
| Jabatan Struktural | : Dosen |
| Jabatan Fungsional | : Lektor |

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut:

| | |
|---|---|
| Nama | : Reynaldi Ganda Dharma Pradana |
| NIM | : 22040002 |
| Program Studi | : Diploma III Teknik Komputer |

Judul TA : RANCANG BANGUN KOPER PINTAR DENGAN PENGHITUNG BERAT DAN PENGENDALIAN MELALUI APLIKASI MOBILE

Dengan ini menyatakan ini dibuat agar dilaksanakan sebagaimana mestinya

Tegal, 5 Februari 2025

| | |
|---|---|
| Mengetahui | |
| Ka Prodi DIII Teknik Komputer, | Dosen Pembimbing I, |
| Ida Afriliana, ST, M.Kom | Safar Dwi Kurniawan, M.Kom |
| NIPY. 12.013.168 | NIPY. 03.021.487 |

## Lampiran 2 Surat Kesediaan Pembimbing 2

**SURAT KESEDIAAN MEMBIMBING TA**

Yang bertanda tangan di bawah ini:

| | |
|---|---|
| Nama | : Lukmanul Khakim, S.Kom., M.Tr.T.,IPP |
| NIPY | : 08.017.343 |
| Jabatan Struktural | : Sub Publikasi |
| Jabatan Fungsional | : Lektor |

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut:

| | |
|---|---|
| Nama | : Reynaldi Ganda Dharma Pradana |
| NIM | : 22040002 |
| Program Studi | : Diploma III Teknik Komputer |

Judul TA : RANCANG BANGUN KOPER PINTAR DENGAN PENGHITUNG BERAT DAN PENGENDALIAN MELALUI APLIKASI MOBILE

Dengan ini menyatakan ini dibuat agar dilaksanakan sebagaimana mestinya

Tegal, 5 Februari 2025

Mengetahui

Ka Prodi DIII Teknik Komputer,

Ida Afriliana, ST, M.Kom
NIPY. 12.013.168

Dosen Pembimbing II,

Lukmanul Khakim, S.Kom., M.Tr.T.,IPP
NIPY. 08.017.343

**Lampiran 3 Bimbingan Proposal**

Lampiran 22 Bimbingan Proposal TA

| IK | P2M | PHB | d.5.1.e.1 |
|---|---|---|---|

**NAMA MAHASISWA :** REYNALDI GANDA DHARMA P

**PEMBIMBING I:** SAFAR KURNIAWAN

*BIMBINGAN PROPOSAL TA*

| No | HARI/ TANGGAL | URAIAN | TANDA TANGAN |
|---|---|---|---|
| | 24 / 12 2024 | Pemaparan Proposal | |
| | 26 / 12/2024 | Revisi Judul | |
| | 28 / 02/2025 | Bimbingan Proposal I II III | |
| | 14 / 3/2025 | Acc proposal | |
| | | Bimbingan Laporan I | |
| | | | |
| | | | |
| | | | |
| | | | |

63

## Lampiran 4 Bimbingan Laporan TA Pembimbing 1

Lampiran 23
Bimbingan Laporan Pembimbing I dan II TA

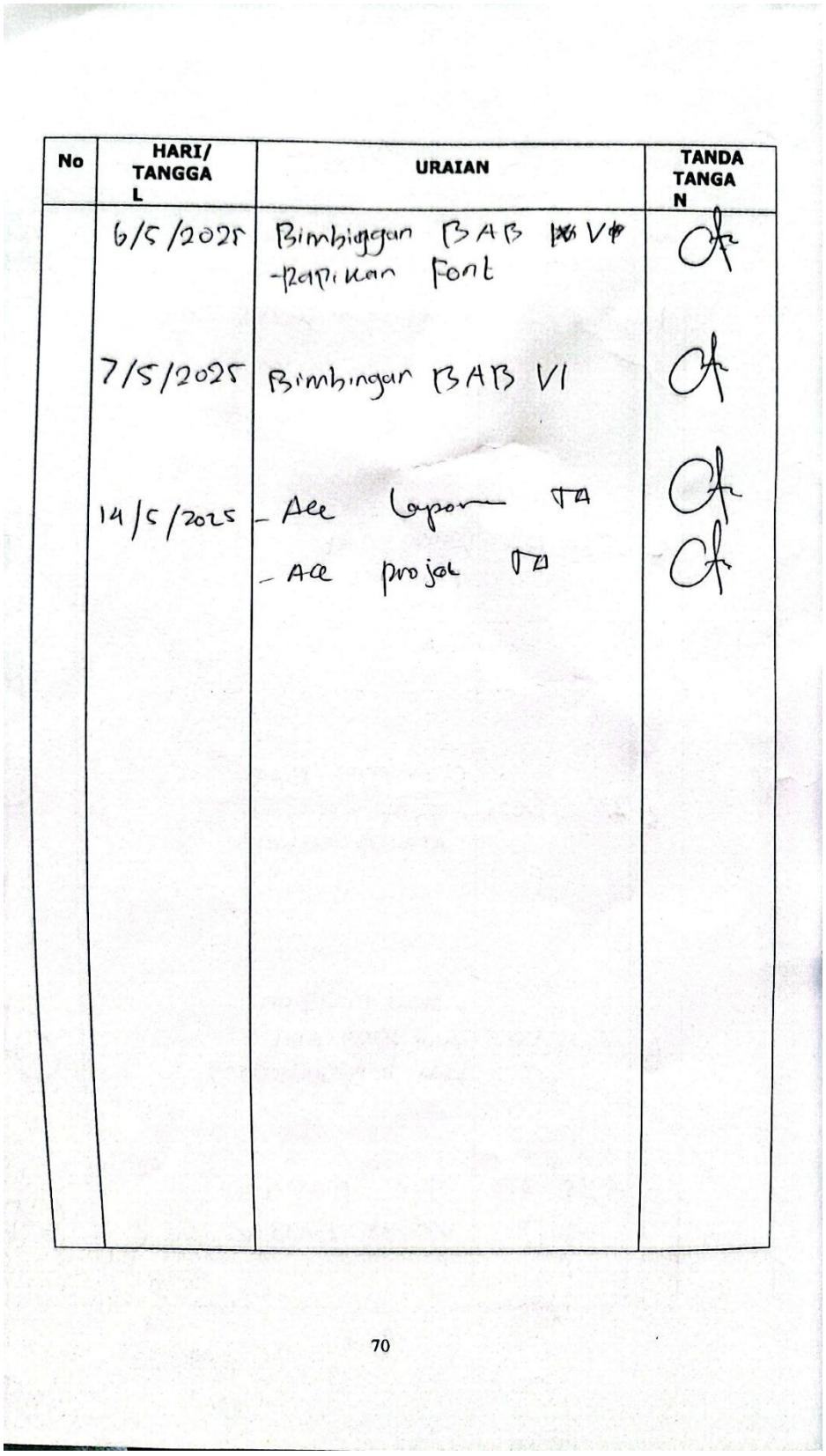
| IK | P2M | PHB | d.5.1.e.1 |
|---|---|---|---|

PEMBIMBING I:.....

*BIMBINGAN LAPORAN TA*

| No | HARI/ TANGGAL | URAIAN | TANDA TANGAN |
|---|---|---|---|
| | 9/4 2025 | Bimbingan laporan I | |
| | 11/4 2025 | Bimbingan Konsep Alat | |
| | 14/2025 | Laporan Bab II | |
| | 16/4 2025 | Laporan Bab III | |
| | | Acc Maju | |

**Lampiran 5 Bimbingan Laporan TA Pembimbing 2**



| No | HARI/ TANGGAL | URAIAN | TANDA TANGAN |
|---|---|---|---|
| | 6/5/2025 | Bimbinggan BAB ~~IV~~ V<br>-Rapikan Font | |
| | 7/5/2025 | Bimbingan BAB VI | |
| | 14/5/2025 | - Acc Laporan TA | |
| | | - Acc projek TA | |

70

**Lampiran 6 Sour Code Program**

```
#include "HX711.h"
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <TinyGPS++.h>
#include <HardwareSerial.h>
#include <time.h>

// Konfigurasi Serial2
#define RXD2 4   // GPS TX ke ESP32 RX (ubah ke GPIO4)
#define TXD2 5   // GPS RX ke ESP32 TX (ubah ke GPIO5)
#define GPS_BAUD 115200  // Ganti ke 115200 jika modul
GPS kamu pakai itu
#define Buzzer 15

// HX711 Instances
HX711 scale1;
HX711 scale2;
HX711 scale3;
HX711 scale4;

// Pin untuk HX711 - Update ke GPIO pins untuk ESP32
const int pinDT_1 = 32;
const int pinSCK_1 = 33;

const int pinDT_2 = 25;
const int pinSCK_2 = 26;

const int pinDT_3 = 27;
const int pinSCK_3 = 14;

const int pinDT_4 = 12;
const int pinSCK_4 = 13;

// Kalibrasi
float calibration_factor1 = 33960.5143 / 340.0;
float calibration_factor2 = 33960.5143 / 340.0;
float calibration_factor3 = 33960.5143 / 340.0;
float calibration_factor4 = 33960.5143 / 340.0;

// Firebase Configuration
#define WIFI_SSID "BAYAR DULU"
#define WIFI_PASSWORD "12345678"
#define FIREBASE_HOST "https://smart-koperv1-default-
rtdb.firebaseio.com/"
```

```
#define FIREBASE_AUTH
"E4i4Hw4TGztRAg7ILis9oA6hqcNbasMO7mrMmRH0"

FirebaseData firebaseData;
FirebaseConfig firebaseConfig;
FirebaseAuth firebaseAuth;

HardwareSerial gpsSerial(2);
TinyGPSPlus gps;
float lat, lang;
int currentHour, currentMinute;
float total_berat = 0;  // Make total_berat a global
variable
unsigned long previousMillisBuzzer = 0;
const long intervalBuzzer = 1000;
bool buzzerState = false;
bool buzzerEnabled = false;

void setup() {
  // Inisialisasi Serial
  Serial.begin(115200);

  // Inisialisasi WiFi
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");
  pinMode(Buzzer, OUTPUT);
  digitalWrite(Buzzer, LOW);
  configTime(); // Konfigurasi waktu NTP

  // Inisialisasi Firebase
  firebaseConfig.host = FIREBASE_HOST;
  firebaseConfig.signer.tokens.legacy_token =
FIREBASE_AUTH;
  Firebase.begin(&firebaseConfig, &firebaseAuth);
  Firebase.reconnectWiFi(true);

  // Inisialisasi HX711
  scale1.begin(pinDT_1, pinSCK_1);
  scale2.begin(pinDT_2, pinSCK_2);
  scale3.begin(pinDT_3, pinSCK_3);
  scale4.begin(pinDT_4, pinSCK_4);

  // Kalibrasi HX711
```

```
  scale1.set_scale(calibration_factor1);
  scale2.set_scale(calibration_factor2);
  scale3.set_scale(calibration_factor3);
  scale4.set_scale(calibration_factor4);

  scale1.tare();
  scale2.tare();
  scale3.tare();
  scale4.tare();

  gpsSerial.begin(GPS_BAUD, SERIAL_8N1, RXD2, TXD2);
  Serial.println("GPS + TinyGPS++ Reader Started");

  delay(2000);
}

void configTime() {
  configTime(7 * 3600, 0, "pool.ntp.org",
"time.nist.gov");
  Serial.print("Waiting for NTP time sync: ");
  time_t now = time(nullptr);
  while (now < 8 * 3600 * 2) {
    delay(500);
    Serial.print(".");
    now = time(nullptr);
  }
  Serial.println();

  struct tm timeinfo;
  localtime_r(&now, &timeinfo);
  Serial.print("Current WIB time: ");
  Serial.print(asctime(&timeinfo));
}

float readHX711(HX711 &scale) {
  if (scale.is_ready()) {
    return scale.get_units(10);  // Baca rata-rata dari
10 pembacaan
  }
  return 0.0;
}

void history(float berat) {
  lat = gps.location.lat();
  lang = gps.location.lng();

  // Get the current time
```

```
  time_t now = time(nullptr);
  struct tm timeinfo;
  localtime_r(&now, &timeinfo);
  currentHour = timeinfo.tm_hour;
  currentMinute = timeinfo.tm_min;

  char dateBuffer[11];
  strftime(dateBuffer, sizeof(dateBuffer), "%Y-%m-%d",
&timeinfo);
  String currentDate = String(dateBuffer);
  char timeBuffer[6];
  strftime(timeBuffer, sizeof(timeBuffer), "%H:%M",
&timeinfo);
  String currentTime = String(timeBuffer);
  String path = "/History/" + currentDate + "/" +
currentTime;

  if (Firebase.setFloat(firebaseData, path + "/berat",
berat)) {
    Serial.println("Berat data berhasil disimpan.");
  } else {
    Serial.println("Gagal menyimpan data berat: " +
firebaseData.errorReason());
  }
  if (Firebase.setFloat(firebaseData, path + "/lat",
lat)) {
    Serial.println("Latitude berhasil disimpan.");
  } else {
    Serial.println("Gagal menyimpan latitude: " +
firebaseData.errorReason());
  }
  if (Firebase.setFloat(firebaseData, path + "/lng",
lang)) {
    Serial.println("Longitude berhasil disimpan.");
  } else {
    Serial.println("Gagal menyimpan longitude: " +
firebaseData.errorReason());
  }
  Serial.println("Data berhasil disimpan pada: " +
path);
}

void controlBuzzer() {

  if (total_berat > 32000) {
    digitalWrite(Buzzer, HIGH);
  } else {
```

```
    digitalWrite(Buzzer, LOW);
  }

  if (Firebase.getString(firebaseData,
"/control/buzer")) {
    String status = firebaseData.stringData();

    if (status == "off") {
      buzzerEnabled = false;
      digitalWrite(Buzzer, LOW);
    }
    else if (status == "on") {
      buzzerEnabled = true;
    }
  }

  if (buzzerEnabled) {
    unsigned long currentMillis = millis();

    if (currentMillis - previousMillisBuzzer >=
intervalBuzzer) {
      previousMillisBuzzer = currentMillis;
      buzzerState = !buzzerState;  // toggle state
      digitalWrite(Buzzer, buzzerState ? HIGH : LOW);
    }
  }
}

void loop() {
  // Read GPS Data
  while (gpsSerial.available()) {
    char c = gpsSerial.read();
    Serial.write(c);      // Tampilkan karakter asli
dari GPS
    gps.encode(c);        // Kirim ke TinyGPS++
  }

  // Tampilkan data hanya jika GPS sudah update (fix)
  if (gps.location.isUpdated()) {
    Serial.println("\n===== GPS Data =====");
    Serial.print("Latitude   : ");
Serial.println(gps.location.lat(), 6);
    Serial.print("Longitude  : ");
Serial.println(gps.location.lng(), 6);
    Serial.print("Satellites : ");
Serial.println(gps.satellites.value());
    Serial.print("HDOP       : ");
```

```
Serial.println(gps.hdop.hdop());
    Serial.print("Altitude   : ");
Serial.println(gps.altitude.meters());
    Serial.print("Date       : ");
    Serial.print(gps.date.day()); Serial.print("/");
    Serial.print(gps.date.month()); Serial.print("/");
    Serial.println(gps.date.year());

    Serial.print("Time (UTC) : ");
    Serial.print(gps.time.hour()); Serial.print(":");
    Serial.print(gps.time.minute()); Serial.print(":");
    Serial.println(gps.time.second());

    Serial.println("====================\n");

    // Send GPS Data to Firebase
    String gpsPath = "/Maps";
    if (Firebase.setFloat(firebaseData, gpsPath +
"/lat", gps.location.lat())) {
      Serial.println("Latitude Data Terkirim");
    } else {
      Serial.println("Failed to send Latitude Data");
      Serial.println(firebaseData.errorReason());
    }

    if (Firebase.setFloat(firebaseData, gpsPath +
"/lng", gps.location.lng())) {
      Serial.println("Longitude Data Terkirim");
    } else {
      Serial.println("Failed to send Longitude Data");
      Serial.println(firebaseData.errorReason());
    }
  }

  // Read the load cell values
  float berat1 = readHX711(scale1);
  float berat2 = readHX711(scale2);
  float berat3 = readHX711(scale3);
  float berat4 = readHX711(scale4);

  berat1 = (berat1 < 0 || berat1 <= 10) ? 0 : berat1;
  berat2 = (berat2 < 0 || berat2 <= 10) ? 0 : berat2;
  berat3 = (berat3 < 0 || berat3 <= 10) ? 0 : berat3;
  berat4 = (berat4 < 0 || berat4 <= 10) ? 0 : berat4;

  // Calculate total weight
  total_berat = berat1 + berat2 + berat3 + berat4;
```

```
  if (total_berat < 0 || total_berat <= 10) {
    total_berat = 0;
  }

  Serial.println("Berat 1: " + String(berat1) + "
Gram");
  Serial.println("Berat 2: " + String(berat2) + "
Gram");
  Serial.println("Berat 3: " + String(berat3) + "
Gram");
  Serial.println("Berat 4: " + String(berat4) + "
Gram");
  Serial.println("Total Berat: " + String(total_berat /
1000.0) + " KG");
  Serial.println("-----------------------------");

  // Send weight data to Firebase
  if (Firebase.getString(firebaseData,
"/control/timbangan")) {
    String status = firebaseData.stringData();

    if (status == "off") {
      if (Firebase.setFloat(firebaseData,
"/Loadcell/berat", 0)) {
        Serial.println("Data Tidak Terkirim");
      } else {
        Serial.println("Gagal Terkirim: " +
firebaseData.errorReason());
      }
      delay(1000);
    }
    else if (status == "on") {
      if (Firebase.setFloat(firebaseData,
"/Loadcell/berat", total_berat)) {
        Serial.println("Data Terkirim: " +
String(total_berat) + " Gram");
      } else {
        Serial.println("Gagal Terkirim: " +
firebaseData.errorReason());
      }
      delay(1000);
    }
  } else {
    // Handle Firebase connection failure
    Serial.println("Failed to get status from Firebase:
" + firebaseData.errorReason());
```

```
  }

  history(total_berat);  // Save history to Firebase
  controlBuzzer();  // Control the buzzer
  delay(1000);  // Wait before the next loop
}
```