

LAMPIRAN

Lampiran 1 Surat Kesiediaan Membimbing TA Pembimbing 1

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

Nama : Nurohim, S.ST,M.Kom
NIDN : 0625067701
NIPY : NIPY.09.017.342
Jabatan Struktual : -
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir Mahasiswa

Berikut:

Nama : Muhamad Asnal Aditya Sumarno
NIM : 23040081
Program Studi : D-3 TEKNIK KOMPUTER SEKOLAH VOKASI
Judul TA : SISTEM MONITORING IOT SUHU,
KELEMBAPAN DAN AMONIA PADA
KANDANG PUYUH BERBASIS WEBSITE

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 9 Maret 2026

Dosen Pembimbing I,

Mengetahui
Ka. Prodi DIII Teknik Komputer,



Arif Rakhman, SE, S.Pd, M.Kom
NIPY. 05.016.291

Nurohim, S.ST, M.Kom
NIPY.09.017.342

Lampiran 2 Surat Kesiediaan Membimbing TA Pembimbing 2

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

Nama : Arif Rakhman, SE, S.Pd, M.Kom.

NIDN : 0623118301

NIPY : 05.016.291

Jabatan Struktual : Ka.Prodi D3 Teknik Komputer

Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir Mahasiswa

Berikut:

Nama : Muhamad Asnal Aditya Sumarno

NIM : 23040081

Program Studi : D-3 TEKNIK KOMPUTER SEKOLAH VOKASI

Judul TA : SISTEM MONITORING IOT SUHU,
KELEMBAPAN DAN AMONIA PADA
KANDANG PUYUH BERBASIS WEBSITE

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 9 Maret 2026

Mengetahui

Dosen Pembimbing II,

Ka. Prodi DIII Teknik Komputer,


Arif Rakhman, SE, S.Pd, M.Kom.
NIPY. 05.016.291


Arif Rakhman, SE, S.Pd, M.Kom.
NIPY.05.016.291

Lampiran 3 Surat Observasi

	Universitas Harkat Negeri	Sekolah Vokasi Program Studi D-3 Teknik Komputer
---	--------------------------------------	---

Nomor : 643.03/D-3 T.Komp.HN/VI/2026
Lampiran : -
Perihal : Permohonan Izin Observasi Tugas Akhir (TA)

Kepada Yth.
Kepala Peternakan Burung Puyuh
Di Pangkah RT 04 RW 05 Kabupaten Tegal

Dengan Hormat,
Sehubungan dengan tugas mata kuliah Tugas Akhir (TA) yang akan diselenggarakan di semester V Program Studi Diploma III Teknik Komputer Universitas Harkat Negeri, Maka dengan ini kami mengajukan izin observasi pengambilan data di Peternakan Burung Puyuh yang Bapak / Ibu Pimpin, untuk kepentingan dalam pembuatan produk Tugas Akhir, dengan Mahasiswa sebagai berikut:

No	NIM	NAMA	NO HP
1	23040081	MUHAMAD ASNAL ADITYA SUMARNO	089506067824
2	23040077	AZWAR DAVA KHALID	085742102880

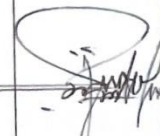
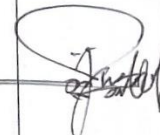
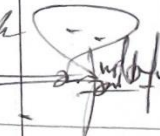
Demikian surat permohonan ini kami sampaikan atas izin dan kerjasamanya kami sampaikan terima kasih.

Tegal, 11 Desember 2025
Ka. Prodi DIII Teknik Komputer
Universitas Harkat Negeri


Arif Rakhman SE, S.Pd, M.Kom
NIPY. 05.016.291

Program Studi D-3 Teknik Komputer, Sekolah Vokasi
Gedung Utama Ruang B.06 Kampus Mataram - Jalan Mataram No 5 Kota Tegal
WhatsApp: +62-85291633073 | Email: komputer@harkatnegeri.ac.id

Lampiran 4 Bimbingan Laporan Pembimbing I TA

Lampiran 22 Bimbingan Proposal TA		IK P2M PHB d.5.1.e.1	
NAMA MAHASISWA : PEMBIMBING I:		BIMBINGAN PROPOSAL TA	
No	HARI/ TANGGAL	URAIAN	TANDA TANGAN
1.	15 April	- Margin masih salah - kata asing (italic) - Gamber (center)	
2.	25 April	- Justify - daftar isi	
3.		- Daftar gambar masih salah	
4.	25 Mei	- Margin masih salah - kata asing (italic) - Justify	
	25 Mei	- Prosedur Penelitian BAB 3	
	26/05 2026	Bab 1, 2, & 3 Acc Lampiran pembimbing 2	

Lampiran 5 Bimbingan Laporan Pembimbing II TA

No	HARI/ TANGGAL	URAIAN	TANDA TANGAN
	.	- Bab 4.	
	,	- Nama gelar masih salah	
		- Sequence Diagram italic	
		Bab 5	
	r	- Daftar isi belum update	
	.	Bab 6	
		- Halaman Pengesahan	
		- Daftar isi Bab IV kosong.	
		- <u>ACL</u> <u>lg</u> <u>ldays</u>	

Lampiran 6 Foto Dokumentasi Observasi



Lampiran 7 *Source Code*

```
<?php

namespace App\Http\Controllers;

use App\Models\SensorData;
use App\Models\SprayerLog;
use App\Models\KipasLog;
use App\Models\LampuLog;
use Illuminate\Http\Request;

class SensorController extends Controller
{
    /**
     * Mengambil data sensor terbaru untuk Dashboard.
     * Prioritas: Manual Log > Sensor Data (otomatis dari
     ESP)
     */
    public function latest()
    {
        $latest = SensorData::latest('id')->first();

        if (!$latest) {
            return response()->json([
                'suhu' => 0,
                'kelembapan' => 0,
                'amonia' => 0,
                'status' => 'menunggu',
            ]);
        }
    }
}
```

```

        'created_at' => now()->format('Y-m-d
H:i:s'),

        'sprayer_active' => false,
        'kipas_active' => false,
        'lampu_active' => false,
        'sprayer_actual' => false,
        'kipas_actual' => false,
        'lampu_actual' => false,

    ]);
}

// FIX: Cek manual log dulu, jika tidak ada
gunakan status otomatis dari SensorData

$latestSprayer = SprayerLog::latest('id')->first();
$latestKipas   = KipasLog::latest('id')->first();
$latestLampu   = LampuLog::latest('id')->first();

// Prioritas: Manual Log (dari tombol) > Sensor
Data (dari ESP otomatis)

$sprayerActive = $latestSprayer
    ? ($latestSprayer->aksi === 'aktif')
    : (bool)$latest->sprayer_active; //
Fallback ke SensorData

$kipasActive = $latestKipas
    ? ($latestKipas->aksi === 'aktif')
    : (bool)$latest->kipas_active; //
Fallback ke SensorData

$lampuActive = $latestLampu

```

```

        ? ($latestLampu->aksi === 'aktif')
        : (bool)$latest->lampu_active;    // Fallback
ke SensorData

return response()->json([
    'suhu' => $latest->suhu,
    'kelembapan' => $latest->kelembapan,
    'amonia' => $latest->amonia,
    'status' => $latest->status,
    'created_at' => $latest->created_at->format('Y-
m-d H:i:s'),
    'sprayer_active' => $sprayerActive,
    'kipas_active' => $kipasActive,
    'lampu_active' => $lampuActive,
    'sprayer_actual' => (bool)$latest-
>sprayer_active,
    'kipas_actual' => (bool)$latest->kipas_active,
    'lampu_actual' => (bool)$latest->lampu_active,
]);
}

/**
 * Mengaktifkan/menonaktifkan sprayer secara manual
dari Dashboard.
 * ESP8266 akan membaca status ini melalui endpoint
espSprayerStatus().
 */
public function toggleSprayer(Request $request)
{

```

```

        $latestSprayer = SprayerLog::latest('id')-
>first();

        $currentlyActive = $latestSprayer
            ? ($latestSprayer->aksi === 'aktif')
            : false;

        $newAction = $currentlyActive ? 'nonaktif' :
'aktif';

        SprayerLog::create([
            'aksi' => $newAction,
            'triggered_by' => 'manual',
        ]);

        return response()->json([
            'success' => true,
            'active' => $newAction === 'aktif',
            'message' => $newAction === 'aktif'
                ? ' Penyemprot Amonia berhasil AKTIF!'
                : ' Penyemprot Amonia berhasil
DINONAKTIFKAN!'
        ]);
    }

/**
     * Mengaktifkan/menonaktifkan kipas secara manual
    dari Dashboard.

     * ESP8266 akan membaca status ini melalui endpoint
    espSprayerStatus().

```

```

*/
public function toggleKipas(Request $request)
{
    $latestKipas = KipasLog::latest('id')->first();

    $currentlyActive = $latestKipas
        ? ($latestKipas->aksi === 'aktif')
        : false;

    $newAction = $currentlyActive ? 'nonaktif' :
    'aktif';

    KipasLog::create([
        'aksi' => $newAction,
        'triggered_by' => 'manual',
    ]);

    return response()->json([
        'success' => true,
        'active' => $newAction === 'aktif',
        'message' => $newAction === 'aktif'
            ? ' Kipas berhasil AKTIF!'
            : ' Kipas berhasil DINONAKTIFKAN!'
    ]);
}

/**
 * Mengaktifkan/menonaktifkan lampu secara manual
 dari Dashboard.

```

```

        * ESP8266 akan membaca status ini melalui endpoint
        espSprayerStatus().

        */

        public function toggleLampu(Request $request)
        {
            $latestLampu = LampuLog::latest('id')->first();

            $currentlyActive = $latestLampu
                ? ($latestLampu->aksi === 'aktif')
                : false;

            $newAction = $currentlyActive ? 'nonaktif' :
            'aktif';

            LampuLog::create([
                'aksi' => $newAction,
                'triggered_by' => 'manual',
            ]);

            return response()->json([
                'success' => true,
                'active' => $newAction === 'aktif',
                'message' => $newAction === 'aktif'
                    ? 'Lampu Pemanas berhasil AKTIF!'
                    : 'Lampu Pemanas berhasil
DINONAKTIFKAN!'
            ]);
        }

```

```

//
=====

// API ENDPOINTS UNTUK ESP8266

//
=====

/**
 * ESP8266 mengirim data sensor ke sini.
 * POST /api/esp/sensor
 * Body: { "suhu": 31.2, "kelembapan": 65.0,
"amonia": 12.5, "kipas_active": false, "lampu_active":
false }
 */
public function espStoreSensor(Request $request)
{
    $validated = $request->validate([
        'suhu' => 'required|numeric',
        'kelembapan' => 'required|numeric',
        'amonia' => 'required|numeric',
        'kipas_active' => 'sometimes|boolean',
        'lampu_active' => 'sometimes|boolean',
        'sprayer_active' => 'sometimes|boolean',
    ]);

    // Penentuan status berdasarkan ambang batas
    bahaya

    $suhu = $validated['suhu'];
    $kelembapan = $validated['kelembapan'];
    $amonia = $validated['amonia'];

```

```

        if ($suhu > 35.0 || $amonia > 25.0 ||
$kelembapan < 40.0 || $kelembapan > 80.0) {
            $status = 'bahaya';
        } elseif ($suhu > 33.0 || $amonia > 20.0 ||
$kelembapan < 50.0 || $kelembapan > 75.0) {
            $status = 'waspada';
        } else {
            $status = 'normal';
        }

        $sensorData = SensorData::create([
            'suhu' => $suhu,
            'kelembapan' => $kelembapan,
            'amonia' => $amonia,
            'status' => $status,
            'kipas_active' =>
$validated['kipas_active'] ?? false,
            'lampu_active' =>
$validated['lampu_active'] ?? false,
            'sprayer_active' =>
$validated['sprayer_active'] ?? false,
        ]);

        return response()->json([
            'success' => true,
            'id' => $sensorData->id,
            'status' => $status,
        ]);
    }

```

```

/**
 * ESP8266 mengecek status sprayer/pompa dari sini.
 * GET /api/esp/sprayer-status
 * Response: { "sprayer_active": true/false }
 *
 * Jika user menekan tombol "Aktifkan Penyemprot"
di dashboard,
 * maka ESP akan membaca sprayer_active = true dan
menyalakan pompa.
 */

public function espSprayerStatus()
{
    $latestSprayer = SprayerLog::latest('id')->first();

    $sprayerActive = $latestSprayer ?
($latestSprayer->aksi === 'aktif') : false;

    $latestKipas = KipasLog::latest('id')->first();

    $kipasActive = $latestKipas ? ($latestKipas->aksi === 'aktif') : false;

    $latestLampu = LampuLog::latest('id')->first();

    $lampuActive = $latestLampu ? ($latestLampu->aksi === 'aktif') : false;

    return response()->json([
        'sprayer_active' => $sprayerActive,
        'kipas_active' => $kipasActive,
        'lampu_active' => $lampuActive,
        'sprayer_ts' => $latestSprayer ?
$latestSprayer->created_at->timestamp : 0,

```

```
        'kipas_ts' => $latestKipas ? $latestKipas->created_at->timestamp : 0,
        'lampu_ts' => $latestLampu ? $latestLampu->created_at->timestamp : 0,
    ];
}
}
```