

LAMPIRAN

Lampiran 1 Surat Kesiediaan Membimbing TA Pembimbing 1

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan dibawah ini:

Nama : Very Kurnia Bakti, M. Kom
NIDN : 0625118301
NIPY : 09.008.044
Jabatan Struktural : Dosen Tetap
Jabatan Fungsional : Lektor

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

Nama : Muhammad Lukman Isma Maulana
Nim : 22040035
Program Studi : DIII Teknik Komputer
Judul TA : PERANCANGAN WEBSITE SISTEM HELPDESK
UNTUK KOORDINASI MASALAH IT
POLITEKNIK HARAPAN BERSAMA

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 18 Maret 2025

Mengetahui,
Ketua Program Studi DIII Teknik Komputer
POLITEKNIK HARAPAN BERSAMA



Very Kurnia Bakti, M. Kom
NIPY. 12.013.168

Pembimbing I

Very Kurnia Bakti, M. Kom
NIPY. 09.008.044

Lampiran 2 Surat Kesiediaan Membimbing TA Pembimbing 2

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan dibawah ini:

Nama : Ahmad Maulana, S.Kom, M.Tr.T
NIDN : 0621099003
NIPY : 11.011.097
Jabatan Struktural : Kepala Bagian Administrasi Akademik
Jabatan Fungsional : Dosen Tetap

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

Nama : Muhammad Lukman Isma Maulana
Nim : 22040035
Program Studi : DIII Teknik Komputer
Judul TA : PERANCANGAN WEBSITE SISTEM HELPDESK
UNTUK KOORDINASI MASALAH IT
POLITEKNIK HARAPAN BERSAMA

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 18 Maret 2025

Mengetahui,
Ketua Program Studi DIII Teknik Komputer
POLITEKNIK HARAPAN BERSAMA

Pembimbing II



Ida Afrilliana ST, M.Kom
NIPY. 12.013.168


Ahmad Maulana, S.Kom, M.Tr.T
NIPY. 11.011.097

Lampiran 3 Surat Permohonan API Wablas

**POLITEKNIK HARAPAN BERSAMA**
The True Vocational Campus

Tegal, 05 Mei 2025

Perihal : Permohonan Akses API Wablas

Yth. Ka. UPT TIK,
di Politeknik Harapan Bersama Tegal

Dengan hormat

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Lukman Isma Maulana
NIM : 22040035
Program Studi : DII Teknik Komputer Politeknik Harapan Bersama
Email : m.lukmanisma@gmail.com

Dengan ini mengajukan permohonan untuk mendapatkan akses API *Wablas* guna mendukung pengembangan sistem tugas akhir saya yang berjudul "PERANCANGAN WEBSITE SISTEM HELPDESK UNTUK KOORDINASI MASALAH IT POLITEKNIK HARAPAN BERSAMA"

API ini akan digunakan untuk mengintegrasikan fitur notifikasi otomatis berbasis WhatsApp, dengan alur sebagai berikut:

1. Notifikasi kepada admin ketika ada laporan baru dari unit,
2. Notifikasi kepada teknisi ketika admin menetapkan tugas penanganan masalah,
3. Notifikasi kepada unit pelapor setiap kali status laporan berubah.

Dengan adanya integrasi API ini, sistem diharapkan mampu meningkatkan efektivitas komunikasi dan mempercepat penyelesaian permasalahan yang dilaporkan melalui sistem Helpdesk.

Sebagai bentuk tanggung jawab, saya bersedia:

- Menggunakan akses API sesuai dengan kebutuhan sistem.
- Menjaga kerahasiaan dan keamanan data yang ditransmisikan melalui API.
- Mematuhi seluruh kebijakan dan prosedur yang berlaku di Sub Bagian TIK Politeknik Harapan Bersama.

 Jl. Mataram No. 9 Kota Tegal 52143, Jawa Tengah, Indonesia.
 (0283)352000

 kemahasiswaan@poltektegal.ac.id
 poltektegal.ac.id



POLITEKNIK HARAPAN BERSAMA
The True Vocational Campus

Demikian surat permohonan ini saya sampaikan. Besar harapan saya agar permohonan ini dapat dikabulkan. Atas perhatian dan kerja samanya, saya ucapkan terima kasih.

Mengetahui,
Ka Prodi DIII Teknik Komputer



Rizka Azzahra ST, M.Kom
NIPY. 07.011.083

Hormat Saya,

Muhammad Lukman Isma Maulana
NIM. 22040035

Lampiran 4 Tabel Perbandingan

No	Aspek	Project	Jurnal 1	Jurnal 2	Jurnal 3	Jurnal 4	Jurnal 5
1	Judul	Sistem HelpDeskTIK untuk Koordinasi dan Resolusi Masalah IT Politeknik Harapan Bersama	Sistem Informasi IT- Helpdesk Universitas Labuhanbatu	Rancang Bangun Sistem Informasi Helpdesk IT Support	Sistem Informasi Helpdesk Berbasis Web USMJ	Sistem Informasi Helpdesk Berbasis Web USMJ	Optimasi Operasional Helpdesk dengan AHP
2	Tujuan	Mempermudah pelaporan & koordinasi permasalahan IT antar unit secara terpusat	Menyediakan sistem pelaporan keluhan pengguna TI	Membantu respon IT support & organisasi data	Menjawab pertanyaan layanan akademik secara cepat	Meningkatkan efisiensi layanan kampus dengan chatbot	Menyeleksi sistem terbaik berdasarkan kriteria
3	Metode	Waterfall	Waterfall	Waterfall	Waterfall	Tidak disebutkan	Tidak disebutkan
4	Platform/Media	Website berbasis CodeIgniter, MySQL, Bootstrap, jQuery, dan API WhatsApp (Wablas)	Website, berbasis ticketing	Website	Website	Chatbot (berbasis RAG)	Aplikasi evaluasi sistem ticketing
5	Fitur Utama	Pelaporan, penggunaan teknis, notifikasi WA, penilaian, komentar, dan dokumentasi	Pelaporan & delegasi teknis	Pelaporan, evaluasi kategori	Layanan akademik, FAQ, penanganan keluhan	Informasi akademik & layanan chatbot	Pemilihan sistem ticketing terbaik
6	Fokus Sistem	Koordinasi laporan lintas unit, transparansi penugasan, notifikasi real-time	Ticketing system, delegasi tugas	Pelaporan internal dan peningkatan efisiensi	Pelayanan informasi akademik dan respon cepat	Kemudahan akses layanan kampus berbasis AI	Seleksi sistem ticketing dengan metode AHP
7	Perbedaan/Pembaruan	Integrasi dengan API WhatsApp (Wablas), notifikasi otomatis, sistem penugasan langsung teknis, dan penilaian user	Sistem masih fokus pada delegasi tugas teknis	Belum fokus pada notifikasi otomatis	Tidak menyediakan penilaian teknis & notifikasi	Fokus pada chatbot, bukan sistem penugasan teknis	Bukan sistem operasional, hanya seleksi alternatif

Lampiran 5 Text Wawancara

- Percakapan dengan Admin TIK

A : Selamat pagi Mba, Saya sedang melakukan penelitian untuk tugas akhir terkait sistem Helpdesk pelaporan IT, dan saya membutuhkan beberapa jawaban terkait tugas akhir saya, apakah saya dapat melakukan wawancara sebentar?

B : Selamat Pagi, silahkan..

A : Mohon izin bertanya, bagaimana proses pelaporan masalah IT dari unit atau prodi ke TIK selama ini?

B : sejauh ini pelaporannya masih sederhana. Biasanya mereka menghubungi langsung lewat WhatsApp, baik ke saya pribadi atau ke WhatsApp Admin yang ada

A : Berarti belum ada sistem pelaporan khusus yang terpusat ya, Pak?

B : Belum ada. Disini masih mengandalkan komunikasi pribadi. Ini sebenarnya sudah sering jadi masalah juga, apalagi kalau sedang banyak laporan masuk.

A : Apa saja kendala yang sering terjadi dari sistem pelaporan seperti itu?

B : Kadang laporannya tidak tersimpan, bisa tenggelam di chat lain, bahkan bisa lupa ditindaklanjuti. Tidak ada catatan khusus, jadi susah kalau mau ditelusuri atau direkap.

A : Kalau begitu, bagaimana proses penugasan ke teknisi dilakukan?

B : Biasanya saya teruskan laporan itu ke teknisi lewat WA juga. Tapi memang terkadang sulit memantau siapa yang sedang mengerjakan tugas apa, atau apakah sudah selesai atau belum.

A : Jika saya buat sistem yang bisa menampung laporan dari unit ke sistem, dan dari sana admin bisa tetapkan teknisi secara langsung, apakah itu akan membantu?

B : Tentu sangat membantu. Tapi apakah sistemnya dapat memberitahu teknisi jika ada tugas masuk?

A : Tentu saja, sistem akan memiliki notifikasi otomatis yang terkirim kepada teknisi ketika admin menetapkan teknisi, tapi mungkin itu hanya akan terjadi jika teknisi sedang membuka tab halaman website.

B : Oh agak susah yahh, mungkin nanti bisa coba koordinasi sama bagian software mas, barangkali nanti ada solusi.

A : Baik Mba.

B : Yang terpenting tetap ada kontrol dari admin di dalam sistem. Dan saya juga ingin sistem ini bisa digunakan untuk evaluasi teknisi. Seperti penilaian setelah masalah selesai.

A : Oke mba, itu sudah saya siapkan. Penilaian berbasis aspek yang dapat diubah.

B : Oke sip, Semoga sistem ini bisa mempermudah kami di TIK dan bantu koordinasi antar unit jadi lebih rapi.

A : Baik, terima kasih mba.

B : Sama-sama mas

- Percakapan dengan Teknisi TIK

A : Selamat siang mas, mohon maaf mengganggu waktunya, saya ingin memberikan beberapa pertanyaan terkait project Tugas Akhir saya yang berkaitan dengan sistem Helpdesk, apakah ada waktu

B : Ya mas silahkan.

A : Untuk pertanyaan pertama terkait bagaimana selama ini teknisi menerima tugas dari admin ya mas?

B : Untuk selama ini admin memberi tugas atau menginfokan jika ada permasalahan masih melalui whatsapp mas,

A : Kalau begitu bagaimana menurut masnya terkait sistem tersebut apakah aman atau cukup?

B : Menurut saya itu kurang aman, karena biasanya dari admin menghubungi melalui whatsapp pribadi, yang tentunya tidak aman, dan pastinya memiliki beberapa resiko chat tersebut tenggelam, kadang juga ada beberapa tugas yang tidak jelas detailnya.

A : Sebenarnya saya sudah mengerjakan beberapa tugas akhir ini, dan beberapa sudah dilakukan seperti detail, penilaian, dan lain-lain, namun untuk notifikasi masih harus memerlukan teknisi agar selalu membuka halaman, dan dari admin menyarankan untuk bertanya ke masnya selaku teknisi software terkait solusi ini.

B : Jika seperti itu permasalahannya masnya bisa menambahkan plugin API WaBlas yang dapat membantu untuk mengirimkan pesan ke teknisi atau

admin jika mendapatkan notifikasi, dari kami sudah ada kok, nantinya tinggal mengajukan surat pengajuan saja.

A : Baik mas terima kasih

B : Dan saya ingin menanyakan apakah nantinya di dalam sistem teknisi atau admin dapat menolak laporan yang dilaporkan atau ditugaskan ke teknisi? Jika belum saya sarankan untuk ditambahkan karena itu lumayan penting agar kita tidak mengambil jobdesk yang salah.

A : Baik mas, untuk saat ini belum ditambahkan tapi mungkin nantinya bakal ditambahkan, sekali lagi saya ucapkan terimakasih mas atas sarannya

B : Iya mas, sama-sama

Lampiran 6 *Source Code* Menu *Controller* Laporan

```
<?php defined('BASEPATH') or exit('No direct script access
allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class Laporan extends MY_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->cek_login(1);
        $this->load->Model('M_Laporan');
    }
    public function meta()
    {
        $dataTeknisi = $this->M_Universal->getMulti('',
'user_teknisi');

        foreach ($dataTeknisi as $item) {
            $totalMasalah = $this->M_Laporan-
>totalMasalah(['id_teknisi' => $item->id, 'masalah.status' => 3],
['end' => date('Y-m-d')]);

            $item->totalMasalah = count($totalMasalah);
        };
        $rangeTgl = $this->input->get('range');
        $tanggal = explode(' to ', $rangeTgl);
        $tglMulai = $tanggal[0];
        $tglSelesai = $tanggal[1];
        if (!empty($rangeTgl)) {
            if (!empty($tglSelesai)) {
                $where = [
                    'date(masalah.created_At) >=' => $tglMulai,
                    'date(masalah.created_At) <=' => $tglSelesai
                ];
            } else {
                $where = [
                    'date(masalah.created_At)' => $tglMulai
                ];
            }
        } else {
            $where = '';
        }
        $data = array(
            "judul" => "laporan Unit",
            "keterangan" => "Contoh Keterangan",
            "halaman" => "laporan",
            "view" => "laporan/laporan",
            "data_masalah" => $this->M_Laporan->konfirmasi($where),
            "dataTeknisi" => $dataTeknisi,
            "rangeTgl" => $rangeTgl,
            "tglMulai" => $tglMulai,
            "tglSelesai" => $tglSelesai
        );
    }
}
```

```

    );
    foreach ($data['data_masalah'] as $masalah) {
        $ruangan = explode('/', $masalah->ruangan);
        $masalah->ruangan = $ruangan[0];
    }
    return $data;
}

public function index()
{
    $this->load->view('template', $this->meta());
}

public function terimaLaporan()
{
    $id = dekrip($this->input->get('id'));

    $update = $this->M_Universal->update(['status' => 1], ['id'
=> $id], 'masalah');

    if ($update) {
        $masalah = $this->M_Universal->getOne(['id' => $id],
'masalah');
        $unit = $this->M_Universal->getOne(['id' => $masalah-
>id_user_unit], 'user_unit');
        $params = [
            'key-gateway' => $this->api_key,
            'token_app' => $this->api_tokenApp,
            'phone' => $unit->no_hp,
            'message' => 'Laporan Anda dengan judul
masalah:\n' . $masalah->masalah . '\n\nTelah diterima oleh
admin.\n\nDetail: ' . base_url('unit/laporan/detail/' .
enkrip($id)),
        ];

        $sendMessage = api_post($this->api_user, $this-
>api_pass, $params, $this->api_host . 'sendSingleMessage', false);
        $sendMessage = json_decode($sendMessage, true);
        if ($sendMessage['status'] == true) {
            notifikasi("success", "Laporan Diterima");
        } else {
            notifikasi("success", "Laporan Diterima, Tapi Gagal
Mengirim Pesan Kepada Unit/Bagian Kerja!");
        };
    } else {
        notifikasi("danger", "Laporan Gagal Diterima");
    }
    echo json_encode(['redirect' => site_url('superadmin/' .
uri(2))]);
}

public function tolakLaporan()
{
    $id = dekrip($this->input->get('id'));
    $message = $this->input->get('message');

    $data = [

```

```

        'status' => 4,
        'message' => $message
    ];
    $update = $this->M_Universal->update($data, ['id' => $id],
'masalah');

    if ($update) {
        $data = [
            'end' => date('Y-m-d H:i:s')
        ];
        $updateKonfirm = $this->M_Universal->update($data,
['id_masalah' => $id], 'konfirmasi');
        if ($updateKonfirm) {
            $masalah = $this->M_Universal->getOne(['id' => $id],
'masalah');
            $unit = $this->M_Universal->getOne(['id' => $masalah-
>id_user_unit], 'user_unit');
            $params = [
                'key-gateway'    => $this->api_key,
                'token_app'      => $this->api_tokenApp,
                'phone'          => $unit->no_hp,
                'message'        => 'Laporan Anda dengan judul
masalah:\n' . $masalah->masalah . '\n\nTelah ditolak oleh
admin.\n\nDetail:      ' . base_url('unit/laporan/detail/' .
enkrip($id)),
            ];
            $sendMessage = api_post($this->api_user, $this-
>api_pass, $params, $this->api_host . 'sendSingleMessage', false);
            $sendMessage = json_decode($sendMessage, true);
            if ($sendMessage['status'] == true) {
                notifikasi("success", "Laporan Ditolak");
            } else {
                notifikasi("success", "Laporan Ditolak, Tapi
Gagal Mengirim Pesan Kepada Unit/Bagian Kerja!");
            };
            } else {
                notifikasi("danger", "Laporan Gagal Ditolak");
            }
        } else {
            notifikasi("danger", "Laporan Gagal Ditolak");
        }
        echo json_encode(['redirect' => site_url('superadmin/' .
uri(2))]);
    }

    public function getEdit()
    {
        $id = dekrip($this->input->get('id'));
        $data['detailLaporan'] = $this->M_Universal->getOne(['id'
=> $id], 'masalah');
        $ruangan = explode('/', $data['detailLaporan']->ruangan);
        if (count($ruangan) == 2) {
            $data['detailLaporan']->ruangan = $ruangan[0];
            $data['detailLaporan']->lokasi_tepat = $ruangan[1];
        }
    }

```

```

        $data['idEnkrip']          = enkrip($data['detailLaporan']->id);
        $data['konfirmasi']       = $this->M_Universal->getOne(['id_masalah' => $id], 'konfirmasi');

        echo json_encode($data);
    }

    public function jadwal()
    {
        $sid = dekrip($this->input->get('id'));
        $masalah = $this->M_Universal->getOne(['id' => $sid], 'masalah');

        $req_id_array = explode(',', $masalah->id_teknisi_req);
        foreach ($req_id_array as $req_id) {
            $data['teknisi'][] = $this->M_Universal->getOne(['id' => $req_id], 'user_teknisi');
        }

        $data['dataJadwal']       = $this->M_Universal->getOneSelect('start', ['id_masalah' => $sid], 'konfirmasi');
        $data['jamSaatIni'] = date('Y-m-d H:i:s');

        echo json_encode($data);
    }

    public function tambah()
    {
        $this->form_validation->set_rules('masalah', 'masalah', 'required');
        $this->form_validation->set_rules('deskripsi', 'Deskripsi Masalah', 'required');
        $this->form_validation->set_rules('jenisMasalah', 'Jenis masalah', 'required');
        if ($this->form_validation->run() == FALSE) {
            $this->notification->error(str_replace("\r\n", "", json_encode(strip_tags(validation_errors()))));
            redirect('superadmin/laporan', 'refresh');
        } else {
            // $idUser = dekrip($this->input->post('id_user'));
            $file = $_FILES['file']['name'];
            $foto = $this->input->post('file');

            $kolom = 'laporan';
            $kolom2 = 'fotoLap';
            $dir = date('Y-m');
            if (is_dir('uploads/' . $kolom) === false) {
                mkdir('uploads/' . $kolom);
                chmod('uploads/' . $kolom, 0755);

                if (is_dir('uploads/' . $kolom . '/' . $kolom2) === false) {
                    mkdir('uploads/' . $kolom . '/' . $kolom2);
                    chmod('uploads/' . $kolom . '/' . $kolom2, 0755);
                }
            }
        }
    }

```

```

        if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
            chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
        }
    } else {
        if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
            chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
        }
    }
} else {
    if (is_dir('uploads/' . $kolom . '/' . $kolom2) ===
false) {
        mkdir('uploads/' . $kolom . '/' . $kolom2);
        chmod('uploads/' . $kolom . '/' . $kolom2,
0755);

        if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
            chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
        }
    } else {
        if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
            chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
        }
    }
}

$this->load->library('upload');
$config['upload_path']          = './uploads/' . $kolom
. '/' . $kolom2 . '/' . $dir;
$config['allowed_types']        = 'png|jpg|jpeg';
$config['max_size']              = 10240; // 10 mb
$config['remove_spaces']        = TRUE;
$config['detect_mime']          = TRUE;
$config['encrypt_name']         = TRUE;

$this->load->library('upload', $config);
$this->upload->initialize($config);

if (!$this->upload->do_upload('file')) {
    $data = [
        'id_user_unit' => $this->user_id,

```

```

        'masalah'      => $this->input->post('masalah'),
        'deskripsi'    => $this->input->post('deskripsi'),
        'tipe_masalah' => $this->input->post('jenisMasalah'),
        'ruangan'      => $this->input->post('ruangan')
        . '/' . $this->input->post('lokasi_tepat'),
    ];
    } else {
        $upload_data = $this->upload->data();

        $data = [
            'id_user_unit' => $this->user_id,
            'masalah'      => $this->input->post('masalah'),
            'deskripsi'    => $this->input->post('deskripsi'),
            'tipe_masalah' => $this->input->post('jenisMasalah'),
            'ruangan'      => $this->input->post('ruangan')
            . '/' . $this->input->post('lokasi_tepat'),
            'foto_laporan' => $dir . '/' .
            $upload_data['file_name']
        ];
    }

    $tambah = $this->M_Universal->insert($data, "masalah");

    $id_masalah = $this->db->insert_id();
    $dataJadwal = [
        'id_masalah' => $id_masalah,
        'start'      => $this->input->post('wPengerjaan'),
        'status'     => 0,
    ];

    $tambahJadwal = $this->M_Universal->insert($dataJadwal,
    "konfirmasi");

    if ($tambah && $tambahJadwal) {
        $admins = $this->M_Universal->getMultiSelect('no_hp', ['level' => 2], 'user_admin');
        $payload = [];
        foreach ($admins as $admin) {
            $payload[] = [
                'phone' => $admin->no_hp,
                'message' => 'Laporan Masuk dari
tester\n\nDetail : ' . base_url('unit/laporan/detail/' .
enkrip($id_masalah)),
            ];
        }

        $params = [
            'key-gateway' => $this->api_key,
            'payload'      => $payload, // array of message
            'token_app'    => $this->api_tokenApp,
        ];
    }
}

```

```

        $sendMessage = api_post($this->api_user, $this->api_pass, $params, $this->api_host . 'sendMessageMultiple', false);
        if ($sendMessage['status'] == true) {
            notifikasi_redirect("success", "Laporan berhasil dikirimkan", 'superadmin/' . uri(2));
        } else {
            notifikasi_redirect("warning", "Laporan berhasil dikirimkan, Tapi Gagal Mengirim Pesan Kepada Admin!", 'superadmin/' . uri(2));
        };
    } else {
        notifikasi_redirect("danger", "Laporan gagal dikirimkan", 'superadmin/' . uri(2));
    }
}

public function edit()
{
    $id = dekrip($this->input->post('id_masalah'));
    $this->form_validation->set_rules('id_masalah', 'Id masalah', 'required');
    $this->form_validation->set_rules('masalah', 'masalah', 'required');
    $this->form_validation->set_rules('deskripsi', 'Deskripsi Masalah', 'required');
    $this->form_validation->set_rules('jenisMasalah', 'Jenis masalah', 'required');
    if ($this->form_validation->run() == FALSE) {
        $this->notification->error(str_replace("\r\n", "", json_encode(strip_tags(validation_errors()))));
    } else {
        $file = $_FILES['file']['name'];
        $foto = $this->input->post('file');

        $kolom = 'laporan';
        $kolom2 = 'fotoLap';
        $dir = date('Y-m');
        if (is_dir('uploads/' . $kolom) === false) {
            mkdir('uploads/' . $kolom);
            chmod('uploads/' . $kolom, 0755);

            if (is_dir('uploads/' . $kolom . '/' . $kolom2) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2);
                chmod('uploads/' . $kolom . '/' . $kolom2, 0755);

                if (is_dir('uploads/' . $kolom . '/' . $kolom2 . '/' . $dir) === false) {
                    mkdir('uploads/' . $kolom . '/' . $kolom2 . '/' . $dir);
                    chmod('uploads/' . $kolom . '/' . $kolom2 . '/' . $dir, 0755);
                }
            }
        }
    }
}

```

```

        } else {
            if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
                chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
            }
        }
    } else {
        if (is_dir('uploads/' . $kolom . '/' . $kolom2) ===
false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2);
            chmod('uploads/' . $kolom . '/' . $kolom2,
0755);

            if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
                chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
            }
        } else {
            if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
                chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
            }
        }
    }

    $this->load->library('upload');
    $config['upload_path'] = './uploads/' . $kolom
. '/' . $kolom2 . '/' . $dir;
    $config['allowed_types'] = 'png|jpg|jpeg';
    $config['max_size'] = 10240; // 10 mb
    $config['remove_spaces'] = TRUE;
    $config['detect_mime'] = TRUE;
    $config['encrypt_name'] = TRUE;

    $this->load->library('upload', $config);
    $this->upload->initialize($config);

    if (!$this->upload->do_upload('file')) {
        $data = [
            'masalah' => $this->input->post('masalah'),
            'tipe_masalah' => $this->input-
>post('jenisMasalah'),
            'deskripsi' => $this->input-
>post('deskripsi'),
            'ruangan' => $this->input->post('ruangan')
. '/' . $this->input->post('lokasi_tepat'),
        ];
    }

```

```

    } else {
        $getfotoLap = $this->M_Universal->getOneSelect('foto_laporan', ['id' => $id], 'masalah');

        if ($getfotoLap != null) {
            unlink('uploads/' . $kolom . '/' . $kolom2 .
                '/' . $getfotoLap->foto_laporan);
        }
        $upload_data = $this->upload->data();

        $data = [
            'masalah' => $this->input->post('masalah'),
            'tipe_masalah' => $this->input->post('jenisMasalah'),
            'deskripsi' => $this->input->post('deskripsi'),
            'ruangan' => $this->input->post('ruangan')
        ];
        $data['lokasi_tepat'] = $this->input->post('lokasi_tepat');
        $data['foto_laporan'] = $this->input->post('foto_laporan');
        $upload_data['file_name'] = $data['foto_laporan'];

        $update = $this->M_Universal->update($data, ['id' => $id], "masalah");

        $dataJadwal = [
            'start' => $this->input->post('wPengerjaan'),
        ];

        $updateJadwal = $this->M_Universal->update($dataJadwal, ['id_masalah' => $id], "konfirmasi");

        if ($update && $updateJadwal) {
            notifikasi_redirect("success", "Update Laporan berhasil", 'superadmin/' . uri(2));
        } else {
            notifikasi_redirect("danger", "Update Laporan gagal", 'superadmin/' . uri(2));
        }
    }
}

public function tugaskan()
{
    $this->form_validation->set_rules('id_masalah', 'Id Masalah', 'required');
    $this->form_validation->set_rules('teknisi', 'Teknisi', 'required');

    if ($this->form_validation->run() == FALSE) {
        $this->notification->error(str_replace("\r\n", "", json_encode(strip_tags(validation_errors()))));
        redirect('superadmin/laporan', 'refresh');
    } else {
        $id = dekrip($this->input->post('id_masalah'));
        $idTeknisi = $this->input->post('teknisi');
        $data = [

```

```

        'id_teknisi' => $idTeknisi,
    ];
    $update      = $this->M_Universal->update($data, ['id'
=> $id], "masalah");

    if ($update) {
        $masalah = $this->M_Universal->getOne(['id' => $id],
'masalah');
        $unit = $this->M_Universal->getOne(['id' => $masalah-
>id_user_unit], 'user_unit');
        $teknisi = $this->M_Universal->getOne(['id' =>
$idTeknisi], 'user_teknisi');
        $params = [
            'key-gateway' => $this->api_key,
            'token_app'    => $this->api_tokenApp,
            'payload'      => [
                [
                    'phone'      => $unit->no_hp,
                    'message'    => 'Laporan anda dengan
judul masalah:\n' . $masalah->masalah . '\n\nTelah ditugaskan
kepada teknisi.\n\nDetail: ' . base_url('unit/laporan/detail/' .
enkrip($id)),
                ],
                [
                    'phone' => $teknisi->no_hp,
                    'message' => 'Laporan dengan judul
masalah:\n' . $masalah->masalah . '\n\nTelah ditugaskan kepada
Anda.\n\nDetail: ' . base_url('teknisi/laporan/detail/' .
enkrip($id)),
                ],
            ],
        ];
        $sendMessage = api_post($this->api_user, $this-
>api_pass, $params, $this->api_host . 'sendMessageMultiple',
false);
        if ($sendMessage['status'] == true) {
            notifikasi_redirect("success", "Teknisi
Ditugaskan", 'superadmin/' . uri(2));
        } else {
            notifikasi_redirect("warning", "Teknisi
Ditugaskan, Tapi Gagal Mengirim Pesan Kepada Teknisi!",
'superadmin/' . uri(2));
        };
        } else {
            notifikasi_redirect("danger", "Teknisi Gagal
Ditugaskan", 'superadmin/' . uri(2));
        }
    }
}
public function batalkan()
{
    $id = dekrip($this->input->get('id'));
    $id_teknisi = dekrip($this->input->get('teknisi'));
    $data = [
        'id_teknisi' => 0,
    ];
}

```

```

        $update      = $this->M_Universal->update($data, ['id' =>
$id], "masalah");

        if ($update) {
            $masalah = $this->M_Universal->getOne(['id' => $id],
'masalah');
            $unit = $this->M_Universal->getOne(['id' => $masalah-
>id_user_unit], 'user_unit');
            $teknisi = $this->M_Universal->getOne(['id' =>
$id_teknisi], 'user_teknisi');
            $params = [
                'key-gateway' => $this->api_key,
                'token_app'   => $this->api_tokenApp,
                'payload'     => [
                    [
                        'phone'      => $unit->no_hp,
                        'message'   => 'Penugasan laporan anda
dengan judul masalah:\n' . $masalah->masalah . '\n\nTelah
dibatalkan, tunggu untuk penugasan selanjutnya.\n\nDetail: ' .
base_url('unit/laporan/detail/' . enkrip($id)),
                    ],
                    [
                        'phone' => $teknisi->no_hp,
                        'message' => 'Laporan yang telah ditugaskan
kepada anda dengan judul masalah:\n' . $masalah->masalah .
'\n\nTelah dibatalkan oleh Admin.\n\nDetail: ' .
base_url('teknisi/laporan/detail/' . enkrip($id)),
                    ],
                ]
            ];
            $sendMessage = api_post($this->api_user, $this-
>api_pass, $params, $this->api_host . 'sendMessageMultiple',
false);
            $sendMessage = json_decode($sendMessage, true);
            if ($sendMessage['status'] == true) {
                echo json_encode('success');
            } else {
                echo json_encode('warning');
            };
        } else {
            echo json_encode('error');
        }
    }
    public function selesai()
    {
        $this->form_validation->set_rules('id_keluhan', 'Id
Keluhan', 'required');
        if ($this->form_validation->run() == FALSE) {
            $this->notification->error(str_replace("\r\n", "",
json_encode(strip_tags(validation_errors()))));
            redirect('superadmin/laporan', 'refresh');
        } else {
            $id = dekrip($this->input->post('id_keluhan'));

            $kolom = 'laporan';
            $kolom2 = 'fotoLapSelesai';

```

```

$dir = date('Y-m');

if (is_dir('uploads/' . $kolom) === false) {
    mkdir('uploads/' . $kolom);
    chmod('uploads/' . $kolom, 0755);

    if (is_dir('uploads/' . $kolom . '/' . $kolom2) ===
false) {
        mkdir('uploads/' . $kolom . '/' . $kolom2);
        chmod('uploads/' . $kolom . '/' . $kolom2,
0755);

        if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
            chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
        }
        } else {
            if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
                chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
            }
        }
    } else {
        if (is_dir('uploads/' . $kolom . '/' . $kolom2) ===
false) {
            mkdir('uploads/' . $kolom . '/' . $kolom2);
            chmod('uploads/' . $kolom . '/' . $kolom2,
0755);

            if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
                chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
            }
        } else {
            if (is_dir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir) === false) {
                mkdir('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir);
                chmod('uploads/' . $kolom . '/' . $kolom2
. '/' . $dir, 0755);
            }
        }
    }

    $this->load->library('upload');
    $config['upload_path'] = './uploads/' . $kolom
. '/' . $kolom2 . '/' . $dir;

```

```

$config['allowed_types']      = 'png|jpg|jpeg';
$config['max_size']          = 10240; // 10 mb
$config['remove_spaces']     = TRUE;
$config['detect_mime']       = TRUE;
$config['encrypt_name']      = TRUE;

$this->load->library('upload', $config);
$this->upload->initialize($config);

if (!$this->upload->do_upload('file')) {
    $data = [
        'message' => $this->input->post('message'),
        'status'  => 3
    ];
} else {
    $upload_data = $this->upload->data();

    $data = [
        'message' => $this->input->post('message'),
        'foto_laporan_selesai' => $dir . '/' .
    $upload_data['file_name'],
        'status' => 3
    ];
}

$update = $this->M_Universal->update($data, ['id' =>
$id], "masalah");

if ($update) {
    $dataJadwal = [
        'end' => date('Y-m-d H:i:s')
    ];

    $updateJadwal = $this->M_Universal->update($dataJadwal, ['id_masalah' => $id], "konfirmasi");

    if ($updateJadwal) {
        $masalah = $this->M_Universal->getOne(['id' =>
$id], 'masalah');
        $id_teknisi = $masalah->id_teknisi;

        $update_stTugas = $this->M_Universal->update(['status' => 0], ['id' => $id_teknisi], 'user_teknisi');
        if ($update_stTugas) {
            $unit = $this->M_Universal->getOne(['id' =>
$masalah->id_user_unit], 'user_unit');
            $admins = $this->M_Universal->getMultiSelect('no_hp', ['level' => 2], 'user_admin');
            $teknisi = $this->M_Universal->getOneSelect('no_hp', ['id' => $id_teknisi], 'user_teknisi');
            $payload = [];
            foreach ($admins as $admin) {
                $payload[] = [
                    'phone' => $admin->no_hp,

```

```

        'message' => 'Laporan dengan judul
masalah:\n' . $masalah->masalah . '\n\nTelah diselesaikan oleh
superadmin.\n\nDetail: ' . base_url('admin/laporan/detail/' .
enkrip($id)),
    ];
}
$msgUnit = [
    'phone' => $unit->no_hp,
    'message' => 'Laporan dengan judul
masalah:\n' . $masalah->masalah . '\n\nTelah diselesaikan oleh
superadmin.\n\nDetail: ' . base_url('unit/laporan/detail/' .
enkrip($id)),
    ];

$msgTeknisi = [
    'phone' => $teknisi->no_hp,
    'message' => 'Laporan yang telah
ditugaskan kepada anda dengan judul masalah:\n' . $masalah->masalah
. '\n\nTelah diselesaikan oleh admin.\n\nDetail: ' .
base_url('teknisi/laporan/detail/' . enkrip($id)),
    ];

array_push($payload, $msgUnit);
array_push($payload, $msgTeknisi);

$params = [
    'key-gateway' => $this->api_key,
    'payload' => $payload, // array of
message objects
    'token_app' => $this->api_tokenApp,
];
$sendMessage = api_post($this->api_user,
$this->api_pass, $params, $this->api_host . 'sendMessageMultiple',
false);

if ($sendMessage['status'] == true) {
    notifikasi_redirect("success", "Laporan
Berhasil Diselesaikan", 'superadmin/' . uri(2));
} else {
    notifikasi_redirect("warning", "Laporan
Berhasil Diselesaikan, Tapi Gagal Mengirim Pesan Kepada Admin dan
Unit!", 'superadmin/' . uri(2));
};
} else {
    notifikasi_redirect("danger", "Laporan Gagal
Diselesaikan", 'superadmin/' . uri(2));
}
} else {
    notifikasi_redirect("danger", "Laporan Gagal
Diselesaikan", 'superadmin/' . uri(2));
}
} else {
    notifikasi_redirect("danger", "Laporan Gagal
Diselesaikan", 'superadmin/' . uri(2));
}
}
}
}

```

```

public function hapus()
{
    $id_masalah = dekrip(uri(4));

    $masalah = $this->M_Universal->getOne(['id' =>
$id_masalah], 'masalah');
    $jadwal = $this->M_Universal->getOne(['id_masalah' =>
$id_masalah], 'konfirmasi');

    unlink(FCPATH . 'uploads/laporan/fotoLap/' . $masalah->foto_laporan);
    unlink(FCPATH . 'uploads/laporan/fotoLapSelesai/' . $masalah->foto_laporan_selesai);
    unlink(FCPATH . 'uploads/laporan/fotoKonfirm/' . $jadwal->foto_konfirmasi);
    $hapusmasalah = $this->M_Universal->delete(['id' =>
$id_masalah], 'masalah');
    $hapusJadwal = $this->M_Universal->delete(['id_masalah'
=> $id_masalah], 'konfirmasi');

    if ($hapusJadwal && $hapusmasalah) {
        notifikasi_redirect("success", "Laporan Terhapus",
'superadmin/' . uri(2));
    } else {
        notifikasi_redirect("danger", "Laporan Tidak Terhapus",
'superadmin/' . uri(2));
    }
}

public function download()
{
    $tglMulai = $this->input->get('startDate');
    $tglSelesai = $this->input->get('endDate');

    if (!empty($tglMulai) && !empty($tglSelesai)) {
        $where = [
            'date(masalah.created_At) >=' => $tglMulai,
            'date(masalah.created_At) <=' => $tglSelesai
        ];
        $nama_file = 'Data Laporan ' . tanggal($tglMulai) . '
- ' . tanggal($tglSelesai);
    } elseif (!empty($tglMulai)) {
        $where = [
            'date(masalah.created_At)' => $tglMulai
        ];
        $nama_file = 'Data Laporan ' . tanggal($tglMulai);
    } else {
        $where = '';
        $nama_file = 'Data Laporan';
    }

    $data_masalah = $this->M_Laporan->konfirmasi($where);

    $spreadsheet = new Spreadsheet();
    $sheet = $spreadsheet->getActiveSheet();

```

```

$sheet->getColumnDimension('A')->setWidth(5);
$sheet->getColumnDimension('B')->setWidth(15);
$sheet->getColumnDimension('C')->setWidth(15);
$sheet->getColumnDimension('D')->setWidth(25);
$sheet->getColumnDimension('E')->setWidth(60);
$sheet->getColumnDimension('F')->setWidth(20);
$sheet->getColumnDimension('G')->setWidth(20);
$sheet->getColumnDimension('H')->setWidth(15);
$sheet->getColumnDimension('I')->setWidth(30);
$sheet->getColumnDimension('J')->setWidth(35);
$sheet->getColumnDimension('K')->setWidth(35);
$sheet->getColumnDimension('L')->setWidth(40);
$sheet->getColumnDimension('M')->setWidth(40);
$sheet->getColumnDimension('N')->setWidth(40);

$sheet->mergeCells('A1:N1');
$sheet->mergeCells('A2:N2');
$sheet->mergeCells('A3:N3');
$sheet->mergeCells('A4:N4');

$sheet->setCellValue('A1', 'LAPORAN HELPPDESK');

$sheet->getStyle('A1:N1')->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);

$sheet->getStyle('A2:N2')->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);

$sheet->getStyle('A3:N3')->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);

$sheet->getStyle('A4:N4')->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);

$sheet->getStyle('A5:N5')->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);
$sheet->getStyle('A5:N5')->getFont()->setBold(true);

$sheet->setCellValue('A5', 'No');
$sheet->setCellValue('B5', 'Unit');
$sheet->setCellValue('C5', 'Tipe');
$sheet->setCellValue('D5', 'Masalah');
$sheet->setCellValue('E5', 'Deskripsi');
$sheet->setCellValue('F5', 'Ruangan');
$sheet->setCellValue('G5', 'Status');
$sheet->setCellValue('H5', 'Teknisi');
$sheet->setCellValue('I5', 'Message');
$sheet->setCellValue('J5', 'Tanggal Mulai');
$sheet->setCellValue('K5', 'Tanggal Selesai');
$sheet->setCellValue('L5', 'Loto Laporan');
$sheet->setCellValue('M5', 'Foto Selesai (Teknisi)');
$sheet->setCellValue('N5', 'Foto Selesai (Unit)');

$no = 1;

```

```

$x = 6;

foreach ($data_masalah as $hasil) {

    if ($hasil->status == 0) {
        $status = 'Menunggu';
    } else if ($hasil->status == 1 && $hasil->id_teknisi !=
0) {
        $status = 'Menunggu Teknisi';
    } else if ($hasil->status == 1) {
        $status = 'Diterima';
    } else if ($hasil->status == 2) {
        $status = 'Dikerjakan';
    } else if ($hasil->status == 3) {
        $status = 'Selesai';
    } else {
        $status = 'Ditolak';
    }

    $penilaian = $this->M_Universal->getOne(['id_masalah'
=> $hasil->id], 'rating');

    $sheet->setCellValue('A' . $x, $no++);
    $sheet->setCellValue('B' . $x, nama_username($hasil-
>id_user_unit, 3));
    $sheet->setCellValue('C' . $x, $hasil->type_masalah ==
1 ? 'Software' : 'Hardware');
    $sheet->setCellValue('D' . $x, $hasil->masalah);
    $sheet->setCellValue('E' . $x, $hasil->deskripsi);
    $sheet->setCellValue('F' . $x, $hasil->ruangan ? $hasil-
>ruangan : 'Tidak Ada Ruangan');
    $sheet->setCellValue('G' . $x, $status);
    $sheet->setCellValue('H' . $x, $hasil->id_teknisi ?
nama_user($hasil->id_teknisi, 4) : 'Belum Ditugaskan');
    $sheet->setCellValue('I' . $x, $hasil->message);
    $sheet->setCellValue('J' . $x, tanggalLengkap($hasil-
>start));
    $sheet->setCellValue('K' . $x, $hasil->end ?
tanggalLengkap($hasil->end) : 'Belum Selesai');
    $sheet->setCellValue('L' . $x, $hasil->foto_laporan ?
base_url('uploads/laporan/fotoLap/') . $hasil->foto_laporan :
'Tidak Ada Foto');
    $sheet->setCellValue('M' . $x, $hasil-
>foto_laporan_selesai ?
base_url('uploads/laporan/fotoLapSelesai/') . $hasil-
>foto_laporan_selesai : 'Tidak Ada Foto Selesai');
    $sheet->setCellValue('N' . $x, $penilaian->foto ?
base_url('uploads/laporan/fotoPenilaian/') . $penilaian->foto :
'Tidak Ada Foto Penilaian');

    $sheet->getStyle('D' . $x)->getAlignment()-
>setWrapText(true);
    $sheet->getStyle('E' . $x)->getAlignment()-
>setWrapText(true);
    $sheet->getStyle('H' . $x)->getAlignment()-
>setWrapText(true);

```

```

        $sheet->getStyle('I' . $x)->getAlignment()-
>setWrapText(true);
        $sheet->getStyle('L' . $x)->getAlignment()-
>setWrapText(true);
        $sheet->getStyle('M' . $x)->getAlignment()-
>setWrapText(true);
        $sheet->getStyle('N' . $x)->getAlignment()-
>setWrapText(true);

        $sheet->getStyle('A' . $x . ':C' . $x)->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);
        $sheet->getStyle('F' . $x . ':H' . $x)->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);
        $sheet->getStyle('J' . $x . ':K' . $x)->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);
        $sheet->getStyle('A' . $x . ':N' . $x)->getAlignment()-
>setVertical(\PhpOffice\PhpSpreadsheet\Style\Alignment::VERTICAL_CENTER);
        $x++;
    }

    $sheet->getStyle('A' . $x)->getAlignment()-
>setHorizontal(\PhpOffice\PhpSpreadsheet\Style\Alignment::HORIZONTAL_CENTER);

    $writer = new Xlsx($spreadsheet);

    header('Content-Type: application/vnd.ms-excel');
    header('Content-Disposition: attachment;filename="' .
$nama_file . '.xlsx"');
    header('Cache-Control: max-age=0');

    $writer->save('php://output');
}
}

```

Lampiran 7 Foto Dokumentasi

